PROBABILISTIC ANALYSIS

FOR FATIGUE STRENGTH DEGRADATION OF MATERIALS

IN-39-CR

185499

P-133

Prepared by:

Lola Boyce, Ph. D., P. E.

Annual Report
of Project Entitled
Development of Advanced Methodologies
for Probabilistic Constitutive Relationships
of Material Strength Models

NASA Grant No. NAG 3-867

Prepared for:

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Lewis Research Center
Cleveland, Ohio 44135

(NASA-CR-182844) PROBABILISTIC ANALYSIS FOR FATIGUE STRENGTH DEGRADATION OF MATERIALS Annual Report (Texas Univ.) 133 p CSCL 20K

N89-15434

Unclas

G3/39 0185499

The Division of Engineering
The University of Texas at San Antonio
San Antonio, TX 78285
January, 1989

PROBABILISTIC ANALYSIS

FOR FATIGUE STRENGTH DEGRADATION OF MATERIALS

Prepared by:

Lola Boyce, Ph. D., P. E.

Annual Report
of Project Entitled
Development of Advanced Methodologies
for Probabilistic Constitutive Relationships
of Material Strength Models

NASA Grant No. NAG 3-867

Prepared for:

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Lewis Research Center
Cleveland, Ohio 44135

The Division of Engineering
The University of Texas at San Antonio
San Antonio, TX 78285

January, 1989

TABLE OF CONTENTS

PREFACE

The University of Texas at San Antonio (UTSA) is a relatively new university. It was established in 1969 and opened for classes in 1973. As the only comprehensive public university serving the nation's ninth largest city, it was and is vital to San Antonio and the entire South Texas Region. In 1983, just seven years ago, an undergraduate engineering program was established at UTSA with the support of the community and its leaders. Today, all three undergraduate engineering programs are ABET accredited and serve about 800 students, a significant percentage of whom are Hispanic. The future includes a new engineering building, providing new laboratory facilities and equipment, together with offices and laboratories, planned to open in January, 1991. Furthermore, a graduate program is planned at both M.S. and Ph.D. levels, and it is hoped that the first Master's Degree students will be able to enroll in Fall, 1989.

Naturally, the engineering research environment is just developing at UTSA. Now, thanks in great measure to the UT System support and this ongoing NASA grant, good progress is being made. Specifically, the purchase of a UT System CRAY-XM/P in March, 1986 and a second one in December, 1988 has provided a world-class analytical and numerical research environment not ordinarily available to a new university. As a result the UTSA Supercomputer Network Research Facility (SNRF) was developed by the principal investigator, Dr. Lola Boyce. This has allowed the successful completion of this research project, the first of its kind at UTSA.

This NASA research grant has allowed two Mechanical Engineering students, Thomas Lovelace and Callie Scheidt, to work directly with the principal investigator, Dr. Boyce, providing them with a quality research experience they would otherwise probably not have had. Both students have expressed an interest in continuing their educations at the graduate level.

In conclusion, and in view of the significant accomplishments in fundamental research, enhancement of the engineering research environment at UTSA, and direct support of Mechanical Engineering students, it is hoped that the proposed extension of this grant will receive favorable consideration at NASA. The principal investigator sincerely thanks NASA for funding this first year grant.

ABSTRACT

This report presents the results of the first year of effort of a program of research conducted for NASA-LeRC by The University of Texas at San Antonio (UTSA). The research included development of methodology that provides a probabilistic treatment of lifetime prediction of structural components of aerospace propulsion systems subjected to fatigue. Material strength degradation models, based on primitive variables, include both a fatigue strength reduction model and a fatigue crack growth model. Linear elastic fracture mechanics is utilized in the latter model. Probabilistic analysis is based on simulation, and both maximum entropy and maximum penalized likelihood methods are used for the generation of probability density functions. The resulting constitutive relationships are included in several computer programs, RANDOM2, RANDOM3 AND RANDOM4. These programs determine the random lifetime, of an engine component, in mechanical load cycles, to reach a critical fatigue strength or crack size. The material considered was a cast nickel base-superalloy, one typical of those used in the Space Shuttle Main Engine (SSME).

LIST OF FIGURES

LIST OF TABLES

1.0 INTRODUCTION

This report presents the results of the first year effort of a research program entitled "Development of Advanced Methodologies for Probabilistic Constitutive Relationships of Material Strength Models." This research is sponsored by the National Aeronautics and Space Administration-Lewis Research Center (NASA-LeRC). The principal investigator is Dr. Lola Boyce, Associate Professor of Mechanical Engineering, The University of Texas at San Antonio (UTSA). The objective of the research program is the development of methodology that provides a probabilistic treatment of lifetime prediction of structural components of aerospace propulsion systems subjected to fatigue.

Two material strength degradation models, based on primitive variables were developed as part of this first year effort: a fatigue crack growth model and a fatigue strength reduction model. The former model utilizes principles of linear elastic fracture mechanics while the latter is, recently developed at NASA-LeRC, quantifies the reduction of strength under cyclic loading, including elevated temperature treatment. Probabilistic analysis is based on simulation, and both maximum entropy and maximum penalized likelihood methods are used for the generation of probability density functions that predict the random lifetime of a material typical of those used in the Space Shuttle Main Engine (SSME), namely a cast nickel base-superalloy.

The resulting constitutive relationships are included in several computer programs, RANDOM2, RANDOM3, and RANDOM4. The programs were developed using both the NASA-LeRC and UTSA Supercomputer Network Research Facility (SNRF) Cray X-MP. New versions of the program accompany this report (see enclosed floppy disk), utilizing the new IMSL Ver. 10 subroutines. Thus, these new versions of the programs will execute on the current NASA-LeRC supercomputer facilities. Also the floppy disk contains sample problems to verify program performance at NASA-LeRC.

Finally, a sensitivity study was carried out for the fatigue strength reduction model for the case of a relatively high mean stress and a relatively low constant amplitude alternating stress at failure. In addition to varying the stresses, the effect of temperature was also considered. A paper was produced documenting much of the effort of this first year research program. This paper is entitled "Probabilistic Constitutive Relationships for Cyclic Material Strength Models", by L. Boyce and C.C. Chamis. It was presented at the 29th Structures, Structural Dynamics and Materials Conference, Williamsburg, VA, April, 1988 and is published in the Proceedings. It has also been submitted to the AIAA Journal of Propulsion and Power.

2.0 FATIGUE CRACK GROWTH MODEL

2.1 Background

Fatigue crack growth data are usually presented as cycles, N, to reach a particular crack length, a. The initial crack size is a_i. It is generally accepted that under constant amplitude alternating stress, fatigue crack growth can be related to stress intensity through a first order differential equation[1]

$$da/dN = C(\Delta K)^m \tag{1}$$

where C is a material parameter, m is a material property (often a constant) and ΔK is the stress intensity range. Stress intensity range is given by

$$\Delta K = Y\Delta\sigma\sqrt{\pi a}$$

where Y is a constant dependent upon component and crack geometry and $\Delta\sigma$ is the constant amplitude alternating stress. Therefore, equation (1) can be written as

$$da/dN = C(Y\Delta\sigma\sqrt{\pi a}\,)^m$$

or,

$$da/dN = C\ Y^m \Delta\sigma^m \pi^{m/2} a^{m/2}. \tag{2}$$

Equation (2) can be integrated, from the initial crack length, a_i, to the final crack length, a_f, to yield N, the number of cycles. The result is

$$N = \frac{1}{CY^m \pi^{m/2} \Delta\sigma^m} \left[\frac{a_f^{-m/2+1} - a_i^{-m/2+1}}{-m/2 + 1} \right] \tag{3}$$

Thus, equation (3) gives the "cycles to reach a given crack length."

Metallurgical evidence indicates that casting pores play a significant role in the high-cycle fatigue life of cast nickel base-superalloys, especially at high temperatures.[2] The location and size of these fatigue crack-initiating pores vary greatly from one aerospace propulsion system component to another. This accounts for the large variability in fatigue life and leads to consideration of fatigue crack growth as a random phenomenon.

Fatigue life directly relates to casting pore size, and pore size can be used to determine initial crack size, a_i. Thus, utilizing principles of both probabilistic analysis and fatigue crack growth, a quantative probabilistic constitutive relationship between fatigue life and fracture mechanics parameters can be developed. Using the "randomized equation" approach, the fatigue crack growth model, given by equation (3) has the following form:

$$N = f(C,m,\Delta\sigma,a_i,a_f,Y) \tag{4}$$

or, in general,

$$N = f(X_i),\ i = 1,....,6, \tag{5}$$

where the X_i are the six independent variables in equations (3) and (4). Equation (3) is "randomized" by assuming the first four variables in equation (4) to be random. Assuming a small crack in a relatively large component leads to assuming $Y = 1.0$, a deterministic value. A deterministic final crack size was chosen since experimental evidence indicated that it was relatively unimportant.[1]

Probabilistic analysis, via simulation, yields the distribution of the dependent random variable, cycles, N. A probability density function (p.d.f.) of cycles is generated using the maximum penalized likelihood method. Maximum penalized likelihood generates the p.d.f. estimate using the method of maximum likelihood together with a penalty function to smooth it.[3]

2.2 RANDOM2 Computer Program

A FORTRAN computer program for the fatigue crack growth model, called RANDOM2, was written using the above-described probabilistic methodology and the constitutive relationship expressed in equation (3). Although the four independent random variables could have any distribution, this initial program provided for normal or lognormal only.

A complete Users Manual for RANDOM2 is contained in Appendix 1. Also, a disk containing a new version of RANDOM2 and a sample problem accompanies this report. The new version of RANDOM2, documented in the Users Manual, uses the new ISML, Ver. 10 subroutines and provides for parameter input from an input file.

3.0 FATIGUE STRENGTH REDUCTION MODEL

3.1 Background

Fatigue strength data are usually presented as cycles to failure for each of several stress amplitudes, the familiar S-N diagram. Results indicate that for lower stress amplitudes the cycles (or time) to failure increases. Thus, a power curve fit through the data yields a monotonically decreasing curve. In general, this curve is represented as

$$S = [N/C']^{-1/m'} \tag{6}$$

where the primitive variables in this equation are as follows: S is the applied constant amplitude alternating stress at failure or fatigue strength, N is number of cycles, C' is a material parameter that varies from specimen to specimen and m' is a material constant.[4] Equation (6) can be written in terms of "cycles to reach a given fatigue strength" as

$$N = C'S^{-m'} \tag{7}$$

Recently another fatigue strength reduction model has been proposed that takes into account the effect of temperature as well as other parameters that affect strength.[5] The general form of the constitutive relationships for this model is applied to the constituents of high temperature composite materials. Specifically, it is applied herein for the case of a single material constituent. The mechanical property of interest is fatigue strength which is expressed in terms of primitive variables,including the general categories of temperature, mechanical cycles and mean stress. For these catagories, the relationship becomes

$$\frac{S}{S_o} = \left[\frac{T_F - T}{T_F - T_o}\right]^n \left[\frac{S_F - \sigma}{S_F - \sigma_o}\right]^m \left[\frac{\log N_{MF} - \log N_M}{\log N_{MF} - \log N_{MO}}\right]^q \tag{8}$$

where S is the applied constant amplitude alternating stress at failure (fatigue strength) at current (or operating) temperature, T, mean stress, σ, and mechanical cycle, N_M. S_O is fatigue strength at reference temperature, T_O (usually room temperature), reference mean stress (or residual stress), σ_O, and reference mechanical cycle, N_{MO}. Also, T_F is the final or melting temperature of the material, S_F is the final or tensile strength of the material, and N_{MF} is the final mechanical cycle or lifetime. Empirical parameters, n, m, and q, are determined from available experimental data or estimated from anticipated behavior of the particular product term[6]. Note that the term containing mechanical cycles is expressed in terms of the log of cycles rather than cycles. This formulation is attractive when N_M and N_{MO} are small compared to N_{MF}. The equation may be solved for N_M, or the "cycles to reach a given fatigue strength." The expression is

$$N = 10\exp\left[\log N_{MF} - \left[(\log N_{MF} - \log N_{MO})\left[\frac{S}{S_o\left[\frac{T_F - T}{T_F - T_o}\right]^n\left[\frac{S_F - \sigma}{S_F - \sigma_o}\right]^m}\right]^{1/q}\right]\right] \tag{9}$$

For values typical of a cast nickel base-superalloy subjected to typical loads and temperatures, equation (9) indicates increasing life for decreasing temperature, decreasing tensile mean stress, and decreasing applied alternating stress. It indicates decreasing life for increasing temperature, decreasing compressive mean stress, and increasing applied alternating stress. Therefore, equation (9) predicts observed trends in general.

Probabilistic analysis, via simulation, yields the distribution of the dependent random variable, cycles, N. A probability density function (p.d.f.) of cycles is generated using the maximum penalized likelihood method for RANDOM3. For RANDOM4, a p.d.f. of cycles is generated using the maximum entropy method. Maximum entropy uses Jaynes' principle which says that "the minimally prejudiced distribution is that which maximizes the entropy subjected to the constraints supplied by the given information."[7]

3.2 RANDOM3 and RANDOM4 Computer Programs

FORTRAN computer programs for the fatigue strenth reduction model called RANDOM3 and RANDOM4 were written using the above-described probabilistic methodology and the constitutive relationship expressed in equation (9). Although the thirteen independent random variables could have any distribution, these programs provided for normal or lognormal only.

A complete Users Manual for RANDOM3 and RANDOM4 is contained in Appendix 2. Also, a disk containing new versions of RANDOM3 and RANDOM4 uses the new IMSL, Ver. 10 subroutines and provides for parameter input from an input file.

3.3 Sensitivity Study

The fatigue strength degradation model using the maximum entropy method of p.d.f. generation (RANDOM4) was selected for use in a sensitivity study. A base line problem utilizing a high mean stress (σ = 90 ksi) and a low constant amplitude alternating stress at failure (S = 22.5 ksi) was established. A room temperature (T = 68^{o}F) problem was executed. The input for this problem is given in Table 1 and the output, in the form of a p.d.f. and a c.d.f. is given in Figures 1 and 2. A high temperature (T= 1562^{o}F) base line problem was also selected. Then, for a fixed base line alternating stress at failure, the mean stress was varied above and below the base line value. Both room and high temperatures were selected. Finally, for a fixed base line mean stress, the alternating stress at failure was varied above and below the base line value. Again, room and high temperatures were selected. A summary of the cases studied is given in Table 2.

Conclusions drawn from this sensitivity study are summarized below. Increasing temperature for the same stress conditions reduces lifetime for all cases (see, for example, Figure 3). At room temperature, when mean stress is increased by 10%, lifetime decreases only very slightly. At high temperature, however, when mean stress is increased by 10%, lifetime decreases substantially. Also, at room temperature, when alternating stress is increased by 30%, lifetime decreases only slightly. At high temperatures, however, when alternating stress is increased by 30%, lifetime decreases very substantially. Considering the above points, lifetime is more sensitive to increasing alternating stress, rather than mean stress. This is probably because alternating stress was increased by 30%, whereas mean stress was increased by only 10%.

Table 1 Base line room temperature (RT) problem input, using the fatigue strength reduction model with maximum entropy p.d.f. generation (RANDOM4)

Variable	Distribution Type	Mean	Std. Dev. Value	Std. Dev. % of Mean
T_F (Melting Temp.)	Normal	2732.0 ^oF	82.0	3
S_F (Ult. Tensile Str.)	Lognormal	130.0 ksi	6.5	5
N_{MF} (Log of Final Cycle)	Lognormal	8.0	0.8	10
T_O (Ref. Temp.)	Normal	68.0 ^oF	2.0	3
σ_O (Residual Comp. Stress)	Lognormal	-2.9 ksi	-0.145	5
N_M (Log of Ref. Cycle)	Lognormal	7.0	0.7	10
S_O (Ref. Fatigue Str.)	Lognormal	72.6 ksi	3.6	5
T (Current Temp.)	Normal	68.0 ^oF	2.0	3
σ (Current Mean Stress)	Lognormal	90.0 ksi	4.5	5
S (Current Fatigue Str.)	Lognormal	22.5 ksi	1.125	5
n (Temp. Exponent)	Normal	0.5	0.015	0.3
m (Stress Exponent)	Normal	0.5	0.015	0.3
q (Cycle Exponent)	Normal	0.5	0.015	0.3

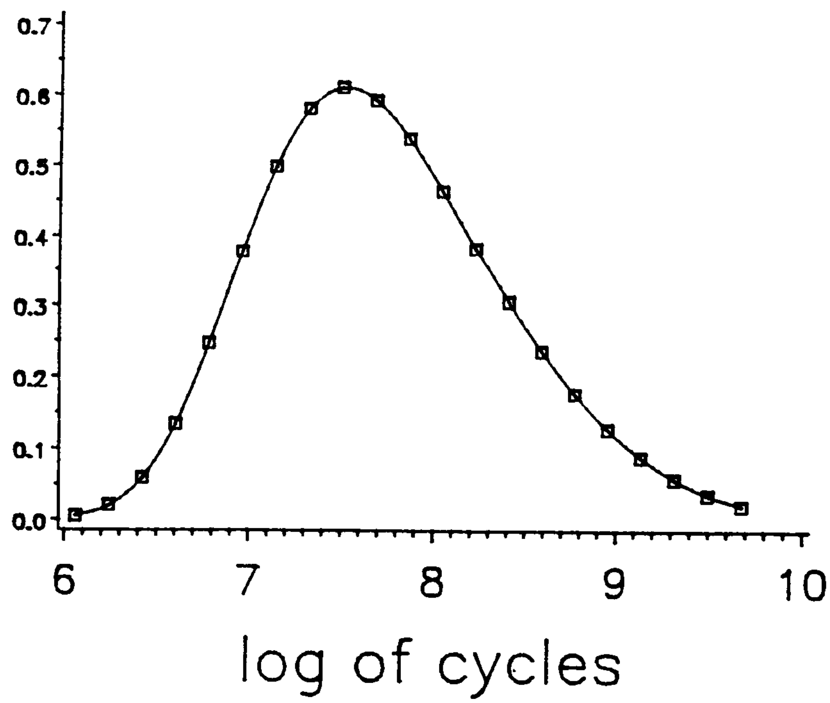

Fig. 1 p.d.f. of base line room temperature (RT) problem

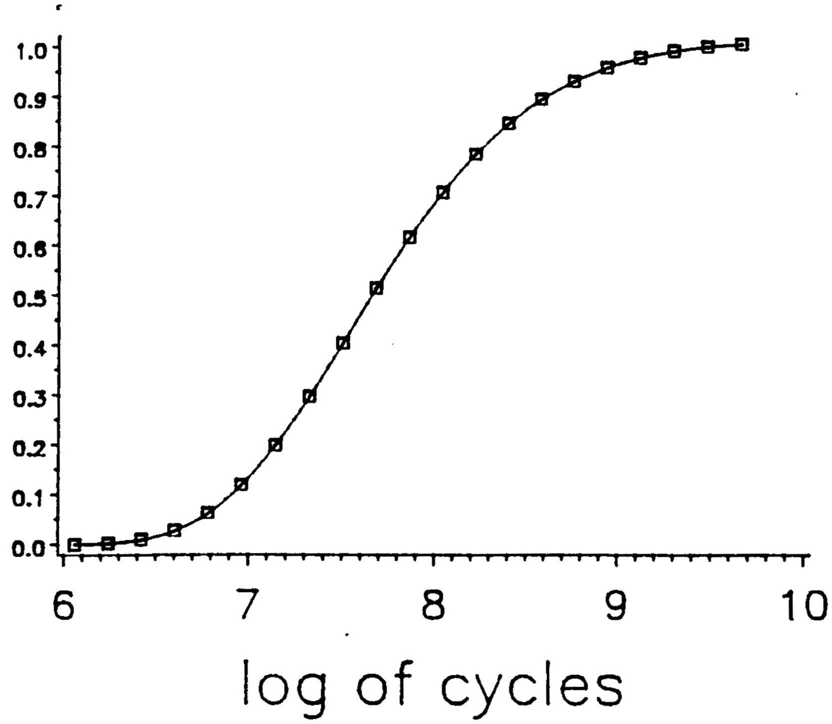

Fig. 2 c.d.f. of base line room temperature (RT) problem

Table 2 Sensitivity study cases for fatigue strength reduction model with maximum entropy p.d.f. generation (RANDOM4)

	σ (ksi)	S (ksi)	T (^{o}F)
	80	22.5	68
Base Line (RT)	90	22.5	68
	100	22.5	68
	80	22.5	1562
Base Line (HT)	90	22.5	1562
	100	22.5	1562
	90	15.0	68
Base Line (RT)	90	22.5	68
	90	30.0	68
	90	15.0	1562
Base Line (HT)	90	22.5	1562
	90	30.0	1562

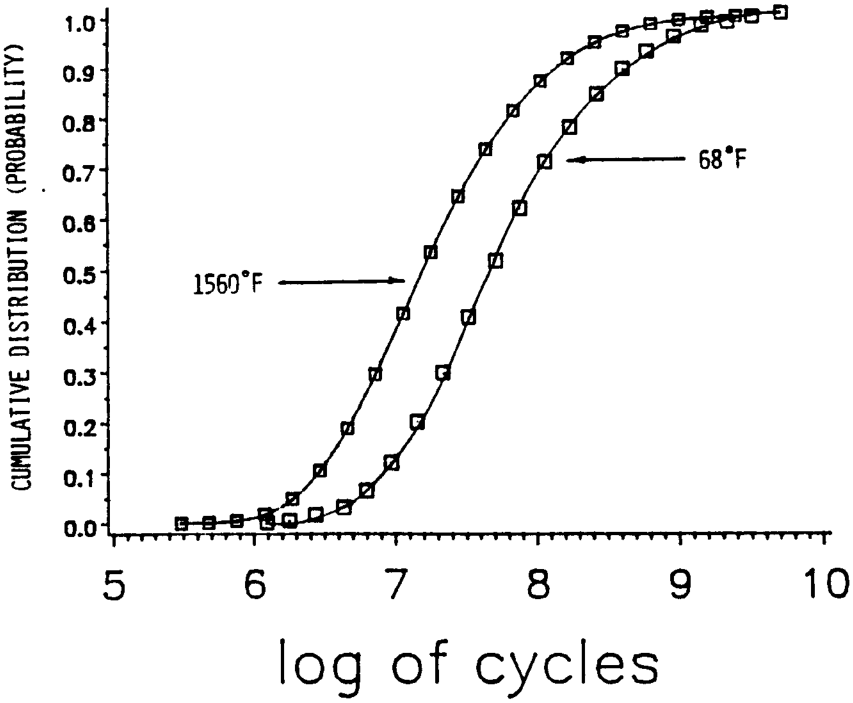

Fig. 3 c.d.f. of base line room temperature (RT) problem compared with c.d.f. of base line high temperature (HT) problem.

4.0 REFERENCES

[1] Kozin, F. and Bogdanoff, J.K., "A Critical Analysis of Some Probabilistic Models of Fatigue Crack Growth," Engineering Fracture Mechanics, Vol. 14, 1981, pp. 55-89.

[2] Hoffeler, W., "High-Cycle Fatigue-Life of the Cast Nickel Base-Superalloys in 738 LC and IN 939," Metallurgical Transactions A, Vol. 13A, July, 1982, pp. 1245-1255.

[3] Scott, D.W., "Nonparametric Probability Density Estimation by Optimization Theoretic Techniques," NASA CR-147763, April, 1976.

[4] Madsen, H.O., "Bayesian Fatigue Life Prediction," Probabilistic Methods in the Mechanics of Solids and Structures, S. Eddwertz and N.C. Lind, Eds., Proceedings of the IUTAM Symposium, Stockholm, Sweden, 1984, pp. 395-406.

[5] Hopkins, D.A. and Chamis, C.C., "A Unique Set of Micromechanics Equations for High Temperature Metal Matrix Composites," NASA TM87154, Nov., 1985.

[6] Chamis, C.C. and Hopkins, D.A., "Thermoviscoplastic Nonlinear Constitutive Relationships for Structural Analysis of High Temperature Metal Matrix Composites," NASA TM 87291, Nov., 1985

[7] Siddall, J.N., "A Comparison of Several Methods of Probabilistic Modeling," Proceedings of the Computers in Engineering Conference, ASME, San Diego, CA, Vol. 4, 1982, pp. 231-238.

5.0 APPENDIX 1

FATIGUE CRACK GROWTH MODEL: RANDOM2 USER MANUAL

FATIGUE CRACK GROWTH MODEL
RANDOM2 USER MANUAL

Prepared by :

Lola Boyce, Ph.D., P.E.
Thomas B. Lovelace

APPENDIX 1
of Annual Report
of Project Entitled
Development of Advanced Methodologies
for Probabilistic Constitutive Relationships
of Material Strength Models

NASA Grant No. NAG 3-867

Prepared for :

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Lewis Research Center
Cleveland, OH 44135

The Division of Engineering
The University of Texas at San Antonio
San Antonio, TX 78285
January, 1989

TABLE OF CONTENTS

1.0 INTRODUCTION

This User Manual documents the FORTRAN program RANDOM2. RANDOM2 is based on fracture mechanics using a probabilistic fatigue crack growth model. It predicts the random lifetime of an engine component to reach a given crack size (see Section 2.0, Theoretical Background).

Included in this Manual are details regarding the theoretical background of RANDOM2, input data instructions and a sample problem illustrating the use of RANDOM2. Appendix A gives information on the physical quantities, their symbols, FORTRAN names, and both SI and U.S. Customary units. Appendix B includes photocopies of the actual computer printout corresponding to the sample problem. Appendices C and D detail the IMSL, Ver. 10 [1], subroutines and functions called by RANDOM2 and a SAS/GRAPH [2] program that can be used to plot both the probability density function (p.d.f.) and the cumulative distribution function (c.d.f.).

2.0 THEORETICAL BACKGROUND

Fatigue crack growth data are usually presented as cycles, N, to reach a particular crack length, a. The initial crack size is a_i. It is generally accepted that under constant amplitude alternating stress, fatigue crack growth can be related to stress intensity through a first order differential equation.[3]

$$da/dN = C(\Delta K)^m \tag{1}$$

where C is a material parameter, m is a material property (often a constant) and ΔK is the stress intensity range. Stress intensity range is given by

$$\Delta K = Y\Delta\sigma\sqrt{\pi a}$$

where Y is a constant dependent upon component and crack geometry and $\Delta\sigma$ is the constant amplitude alternating stress. Therefore, equation (1) can be written as

$$da/dN = C(Y\Delta\sigma\sqrt{\pi a}\,)^m$$

or,

$$da/dN = C\,Y^m\Delta\sigma^m\pi^{m/2}a^{m/2}. \tag{2}$$

Equation (2) can be integrated, from the initial crack length, a_i, to the final crack length, a_f, to yield N, the number of cycles. The result is

$$N = \frac{1}{CY^m\pi^{m/2}\Delta\sigma^m}\left[\frac{a_f^{-m/2+1} - a_i^{-m/2+1}}{-m/2 + 1}\right] \tag{3}$$

Thus, equation (3) gives the "cycles to reach a given crack length."

Metallurgical evidence indicates that casting pores play a significant role in the high-cycle fatigue life of cast nickel base-superalloys, especially at high temperatures.[4] The location and size of these fatigue crack-initiating pores vary greatly from one aerospace propulsion system component to another. This accounts for the large variability in fatigue life and leads to consideration of fatigue crack growth as a random phenomenon.

Fatigue life directly relates to casting pore size, and pore size can be used to determine initial crack size, a_i. Thus, utilizing principles of both probabilistic analysis and fatigue crack growth, a quantitative probabilistic constitutive relationship between fatigue life and fracture mechanics parameters can be developed. Using the "randomized equation" approach, the fatigue crack growth model, given by equation (3) has the following form:

$$N = f(C,m,\Delta\sigma,a_i,a_f,Y) \tag{4}$$

or, in general,

$$N = f(X_i),\ i = 1,....,6, \tag{5}$$

where the X_i are the six independent variables in equations (3) and (4). Equation (3) is "randomized" by assuming the first four variables in equation (4) to be random. Assuming a small crack in a relatively large component leads to assuming $Y = 1.0$, a deterministic value. A deterministic final crack size was chosen since experimental evidence indicated that it was relatively unimportant.[3]

Probabilistic analysis, via simulation, yields the distribution of the dependent random variable, cycles, N. A probability density function (p.d.f.) of cycles is generated using the maximum penalized likelihood method. Maximum penalized likelihood generates the p.d.f. estimate using the method of maximum likelihood together with a penalty function to smooth it.[5]

3.0 INPUT DATA

Data input for RANDOM2 is user friendly and easy to manipulate (see, for example, the file entitled NORMAL.INP, in Section 4.0). The first five lines of input have the same format, namely 2E12.4, and the last two lines differ. The last two lines of input have the formats I3,2X,I3,2X,2E12.4,2X,I3 and I3, respectively. A brief line by line description is given along with an example for each line (Note: the ruler is to aid the user in formatting and is not a part of the input). A table listing the physical quantities, their units and symbols is given in Appendix A.

1. Random Number Generator Seed, ISEED, and Sample Size, NTOT

EXAMPLE:

```
123456789012345678901234567890
                      1          40
```

2. Material Property, RMM

EXAMPLE:

```
123456789012345678901234567890
    28.0E-01     1.4E-01
```

3. Initial Crack Size (Pore Diameter), RAI

EXAMPLE:

```
123456789012345678901234567890
   300.0E-06   45.0E-06
```

4. Material Property, RCC

EXAMPLE:

```
123456789012345678901234567890
    2.20E-11   0.22E-11
```

5. Stress Range, DELSIG

EXAMPLE:

```
123456789012345678901234567890
     6.2E+02     6.2E+01
```

6. The DESPL [1] parameters are NODE, INIT, ALPHA, EPS, MAXIT and are entered in that order as follows:

EXAMPLE:

```
1234567890123456789012345678901234567890
 21     0         50.0E-01     10.0E-05   30
```

7. The DESPL parameter, IOPT, is entered as follows:

EXAMPLE:

```
1234567890
  2
```

4.0 SAMPLE PROBLEM FOR RANDOM2

The objective of this program is to predict the random lifetime, to reach a given crack size for an engine component . The theory is based on fracture mechanics, using a probabilistic fatigue crack growth model (see Section 2.0, Theoretical Background). RANDOM2 input parameters are given in Table A1.1. Note that the first four parameters are random. Their means and standard deviations are input by the user. The last two parameters, A_f and Y, are deterministic and are fixed internally by the program. They are equal to the values shown in Table A1.1.

Table A1.1 RANDOM2 sample problem input (SI units)

FORTRAN Name	Distribution Type	Mean	Standard Deviation (Value)	(% of Mean)
RMM	normal	28.0E-01	1.4E-01	(5%)
AI	lognormal	300.0E-06	45.0E-06	(15%)
RCC	lognormal	2.20E-11	0.22E-11	(10%)
DELSIG	lognormal	6.2E+02	6.2E+01	(10%)
AF	N/A	2.0E-03	N/A	
YY	N/A	1.0	N/A	

The input is entered in the following format in a file entitled NORMAL.INP.

```
1234567890123456789012345678901234567890
           1           40
    28.0E-01      1.4E-01
   300.0E-06     45.0E-06
    2.20E-11     0.22E-11
     6.2E+02      6.2E+01
  21    0       50.0E-01    10.0E-05    30
   2
```

Execution of RANDOM2 (source code entitled NR2.FOR) produces an output file entitled RANDM22 giving intermediate results (see Appendix B). Execution also produces the plotfiles OUT1 and OUT2 (see Appendix B). These files are used to plot the X and Y axes of the probability density function (p.d.f.) and the cumulative distribution function (c.d.f.), respectively, generated by RANDOM2. The plots are drawn from the plotfiles by the SAS/GRAPH graphing program (see Appendix C). These plots for the sample problem are shown in Figures A1.1 and A1.2.

This same sample problem has been reported in Boyce and Chamis.[6] There, however, it utilized U.S. Customary units and an older version of RANDOM2 (IMSL Version 9.2 subroutines).

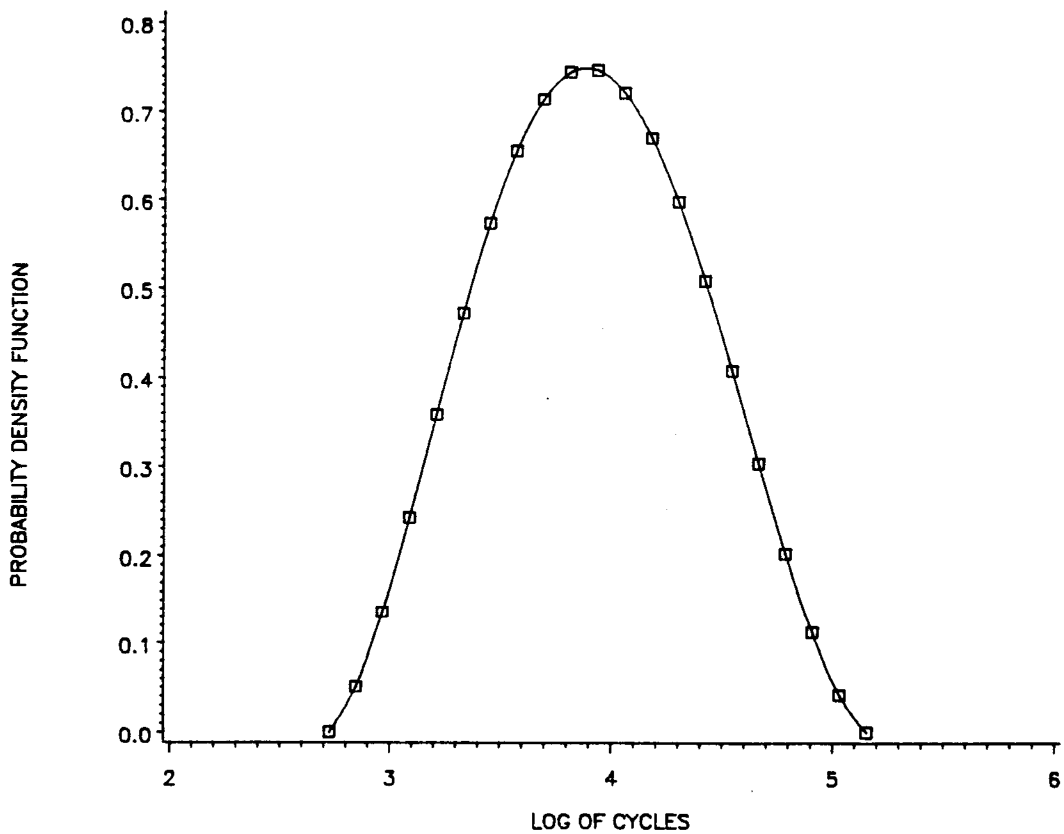

Fig. A1.1 p.d.f. of log of mechanical cycles for fatigue crack growth model, using maximum penalized likelihood.

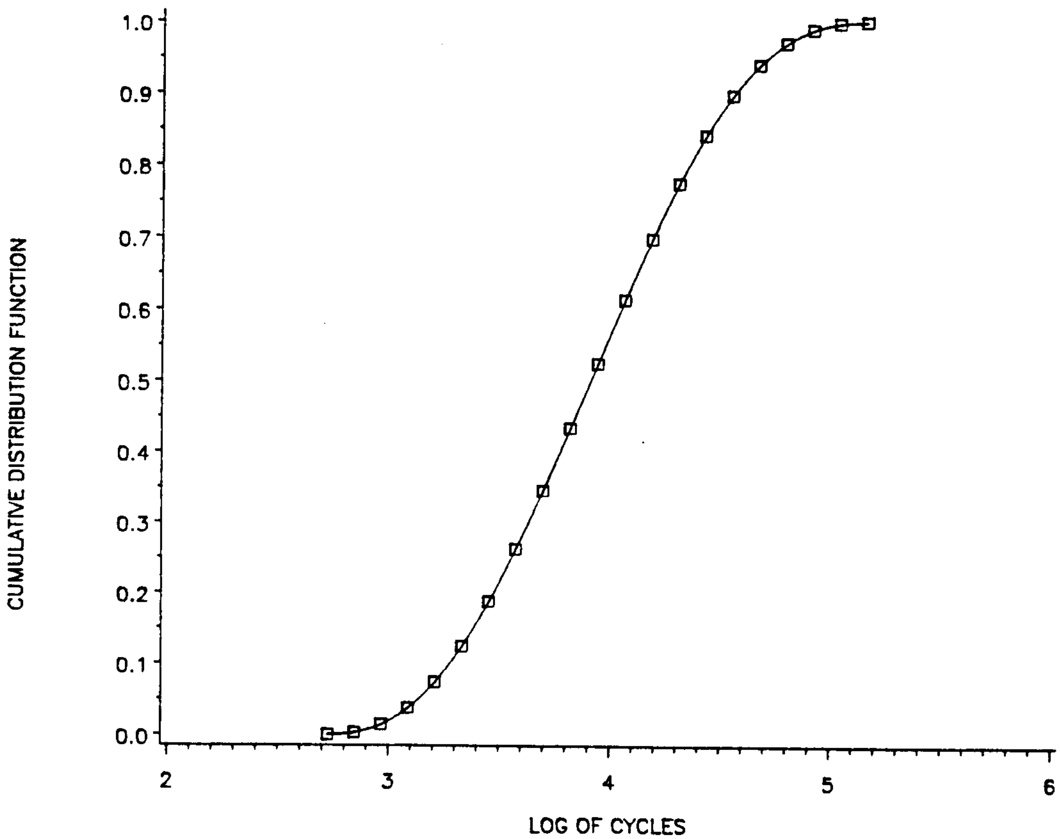

Fig. A1.2 c.d.f. of log of mechanical cycles for fatigue crack growth model, using maximum penalized likelihood.

5.0 REFERENCES

[1] IMSL, "STAT/LIBRARY, FORTRAN Subroutines for Statistical Analysis", Houston, Texas, 1987.

[2] SAS Institute, Inc. SAS/GRAPH User's Guide, Version 5 Edition, Cary, NC: SAS Institute, Inc., 1985, p. 596.

[3] Kozin, F. and Bogdanoff, J.K., "A Critical Analysis of Some Probabilistic Models of Fatigue Crack Growth," Engineering Fracture Mechanics, Vol. 14, 1981, pp. 55-89.

[4] Hoffeler, W., "High-Cycle Fatigue-Life of the Cast Nickel Base-Superalloys in 738 LC and IN 939," Metallurgical Transactions A, Vol. 13A, July, 1982, pp. 1245-1255.

[5] Scott, D.W., "Nonparametric Probability Density Estimation by Optimization Theoretic Techniques," NASA CR-147763, April, 1976.

[6] Boyce, L. and Chamis, C.C., "Probabilistic Constitutive Relations for Cyclic Material Strength Models," Proceedings, 29th Structures, Structural Dynamics and Materials Conference, Williamsburg, VA, 1988.

6.0 APPENDIX A

PHYSICAL QUANTITIES, SYMBOLS, AND UNITS

The physical quantities, their symbols, and units for the fatigue crack growth model are given in the following table.

Table A1.2 Physical quantities, symbols, and units for fatigue crack growth model for RANDOM2

Physical Quantity	Theory Symbol	FORTRAN Name	Units SI	Units U.S.
Material Property	m	RMM	m/cycle/M Pa m	in/cycle/ksi in
Initial Crack Size	A_i	RAI	m	in
Material Property	C	RCC	m/cycle	in/cycle
Alternating Stress	$\Delta\sigma$	DELSIG	M Pa	ksi
Final Crack Size	A_f	AF	m	in
Geometry Dependent Constant	Y	YY	(dimensionless)	

7.0 APPENDIX B

SAMPLE PROBLEM: SOURCE, INPUT AND OUTPUT FILES

```
DDDDDDDDDD   333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333
DDDDDDDDDD   3333333333333333333333333  Digital Equipment Corporation - VAX/VMS Version V4.7  3333333333333333333333333   DDDDDDDD
DDDDDDDDDD   333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333   DDDDDDDD
                                                                                                                   DDDDDDDD

                      DDDD    EEEEE   CCCC   N    N  EEEEE  TTTTT
                      D   D   E      C       N    N  E        T
                      D   D   E      C       NN   N  E        T
                      D   D   EEEE   C       N N  N  EEEE     T
                      D   D   E      C       N  NN   E        T
                      D   D   E      C       N    N  E        T
                      DDDD    EEEEE   CCCC   N    N  EEEEE    T

                       NN        NN  RRRRRRRR        222222
                       NN        NN  RRRRRRRR        222222
                       NN        NN  RR        RR  22      22
                       NN        NN  RR        RR  22      22
                       NNNN      NN  RR        RR          22
                       NNNN      NN  RR        RR          22
                       NN  NN    NN  RRRRRRRR            22
                       NN  NN    NN  RRRRRRRR            22
                       NN    NNNN    RR    RR          22
                       NN    NNNN    RR    RR          22
                       NN        NN  RR      RR      22
                       NN        NN  RR      RR      22
                       NN        NN  RR        RR  2222222222
                       NN        NN  RR        RR  2222222222

              FFFFFFFFFF      000000      RRRRRRRR      ;;;;    77777777
              FFFFFFFFFF      000000      RRRRRRRR      ;;;;    77777777
              FF            00      00    RR      RR    ;;;;          77
              FF            00      00    RR      RR    ;;;;          77
              FF            00      00    RR      RR                  77
              FF            00      00    RR      RR                  77
              FFFFFFFF      00      00    RRRRRRRR      ;;;;        77
              FFFFFFFF      00      00    RRRRRRRR      ;;;;        77
              FF            00      00    RR  RR        ;;;;      77
              FF            00      00    RR  RR        ;;;;      77
        ....  FF            00      00    RR    RR        ;;    77
        ....  FF            00      00    RR    RR        ;;    77
        ....  FF              000000      RR      RR    ;;    77
        ....  FF              000000      RR      RR    ;;    77
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
File _DUA0:[]NR2.FOR;7 (648,66,0), last revised on 29-NOV-1988 11:19, is a 43 block sequential file owned by UIC [DECNET].  The
records are variable length with implied (CR) carriage control.  The longest record is 72 bytes.

Job NR2 (1814) queued to TERM$LA120A on 29-NOV-1988 11:21 by user DECNET, UIC [DECNET], under account DECNET at priority 100, starte
on printer LTA4: on 29-NOV-1988 11:21 from queue TERM$LA120A.

DDDDDDDDDD   333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333   DDDDDDDD
DDDDDDDDDD   3333333333333333333333333  Digital Equipment Corporation - VAX/VMS Version V4.7  3333333333333333333333333   DDDDDDDD
DDDDDDDDDD   333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333   DDDDDDDD
```

```
JOB,JN=COMPL,US=USAQ530,RT=60,T=30,MFL=3000000.
ACCOUNT,UPW=LOLAB.
lib,dn=imslib.
CFT77,LIST.
REWIND,DN=$BLD.
BUILD,I=0,OBL=0.
SAVE,DN=$NBL,PDN=NR2BLD,ID=SMBOYCE.
DELETE,PDN=NR2BLD,ID=SMBOYCE,ED=-1.
/EOF
C PARIS-ERDOGAN FATIGUE CRACK GROWTH EQUATION
C RANDOMIZED AND APPLIED TO CRACK LENGTH
C WHERE INITIAL CRACK SIZE IS RANDOM PORE DIAMETER
      INTEGER ISEED,NTOT,M,INIT,NMISS,MAXIT,NODE
      REAL XM,XS,YM,YS,EPS,P,RWKSP(9999),ALPHA
      COMMON/WORKSP/RWKSP
      DIMENSION RMM(10000),RAI(10000),RCC(10000)
      DIMENSION DELSIG(10000),BNDSX(1000)
      DIMENSION XNF(10000),C(10000)
      DIMENSION STAT(9999),DENS(1000),DISTX(1000)
      DIMENSION BNDS(10000),BB(1000),FF(1000)
      EXTERNAL RNLNL,RNSET,RNNOR,DESPL,IWKIN
 1001 FORMAT(5E12.4)
 1002 FORMAT(I12,I12)
 1003 FORMAT(I4,I4)
 1004 FORMAT(I4)
 1009 FORMAT(I3,2X,I3,2X,2E12.4,2X,I3)
 1010 FORMAT(I3)
 1011 FORMAT(2E12.4)
C NORMAL MATERIAL PROPERTY, M
      READ (5,1002)ISEED,NTOT
      WRITE(6,1002)ISEED,NTOT
      READ (5,1011)YM,YS
      WRITE(6,1011)YM,YS
C      YS=0.14
C      YM=2.8
      CALL RNSET(ISEED)
      CALL RNNOR(NTOT,RMM)
      DO 102 I=1,NTOT
      RMM(I)=YS*RMM(I)+YM
  102 CONTINUE
      WRITE(6,2019)
 2019 FORMAT(' MATERIAL PROPERTY, M')
      WRITE( 6,1001) (RMM(I),I=1,NTOT)
C LOGNORMAL INITIAL CRACK SIZE (PORE DIAMETER),AI
      WRITE(6,1002) ISEED,NTOT
      READ (5,1011) XM,XS
      WRITE(6,1011) XM,XS
C      XM  = 300.0E-06
C      XS  = 45.0E-06
      YS = SQRT( LOG(1.0+(XS/XM)**2) )
      YM = LOG(XM) - 0.5*YS**2
      CALL RNSET(ISEED)
      CALL RNLNL(NTOT,YM,YS,RAI)
      WRITE(6,2020)
 2020 FORMAT(' INITIAL CRACK SIZE, AI')
      WRITE( 6,1001) (RAI(I),I=1,NTOT)
C LOGNORMAL MATERIAL PROPERTY, C
      WRITE(6,1002) ISEED,NTOT
      READ (5,1011) XM,XS
      WRITE(6,1011) XM,XS
C      XM  = 2.20E-11
C      XS  = 0.220E-11
      YS = SQRT( LOG(1.0+(XS/XM)**2) )
      YM = LOG(XM) - 0.5*YS**2
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      CALL RNSET(ISEED)
      CALL RNLNL(NTOT,YM,YS,RCC)
      WRITE(6,2021)
 2021 FORMAT(' MATERIAL PROPERTY, C')
      WRITE( 6,1001) (RCC(I),I=1,NTOT)
C LOGNORMAL STRESS RANGE, DELSIG
      WRITE(6,1002)ISEED,NTOT
      READ (5,1011) XM,XS
      WRITE(6,1011) XM,XS
C      XS = 0.62E+02
C      XM = 6.2E+02
      YS = SQRT(LOG(1.0+(XS/XM)**2))
      YM = LOG(XM) - 0.5*YS**2
      CALL RNSET(ISEED)
      CALL RNLNL(NTOT,YM,YS,DELSIG)
      WRITE (6,2022)
 2022 FORMAT (' STRESS RANGE, DELSIG')
      WRITE (6,1001)(DELSIG(I),I=1,NTOT)
C DEFINE DETERMINISTIC PARAMETERS
C PI
      PI = 3.14159265358979323846264 33
C COMPONENT AND CRACK SHAPE PARAMETER, YY
      YY=1.0
C FINAL CRACK SIZE, AF
      AF=2.0E-03
C CALCULATE CYCLES TO REACH CRACK SIZE 2.0E-03M
      DO 101 I=1,NTOT
      XNF1=1.0/(RCC(I)*YY**RMM(I)*PI**(RMM(I)/2.)*DELSIG(I)**
     1RMM(I))
      XNF2=((AF**(1.-RMM(I)/2.)-RAI(I)**(1.-RMM(I)/2.))/
     I(1.-RMM(I)/2.))
      XNF(I)=XNF1*XNF2
C CALCULATE LOG OF CYCLES TO REACH CRACK SIZE 2.0E-03M
      XNF(I)=ALOG10(XNF(I))
  101 CONTINUE
      WRITE(6,2023)
 2023 FORMAT (' LOG OF CYCLES TO REACH CRACK SIZE=2.0E-03M,',/,
     1'GIVEN STRESS MEAN AMPLITUDE=6.2E+02MPA')
      WRITE(6,1001)(XNF(I),I=1,NTOT)
C SORT LOG OF CYCLES
      CALL SORT(XNF,NTOT)
      WRITE (6,2024)
 2024 FORMAT (' SORTED LOG OF CYCLES')
      WRITE (6,1001)(XNF(I),I=1,NTOT)
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
C CALCULATE PDF OF LOG OF CURRENT CYCLES,LOG XNF
      READ(5,1009)NODE,INIT,ALPHA,EPS,MAXIT
      WRITE(6,985)
  985 FORMAT(' DESPL PARAMETERS')
      WRITE(6,1009)NODE,INIT,ALPHA,EPS,MAXIT
      BNDS(1)=XNF(1) - 0.05*XNF(1)
      BNDS(2)=XNF(NTOT) + 0.05*XNF(NTOT)
      WRITE(6,979)BNDS(1),BNDS(2)
  979 FORMAT('BNDS(1),BNDS(2)=',E12.4,1X,E12.4)
      CALL DESPL(NTOT,XNF,NODE,BNDS,INIT,ALPHA,MAXIT,EPS,DENS,STAT,
     1NMISS)
      WRITE(6,980)
  980 FORMAT('PDF OF LOG OF CURRENT CYCLES,LOG XNF,Y AXIS OF PDF PLOT')
      WRITE(6,1001)(DENS(I),I=1,NODE)
      WRITE(6,981)
  981 FORMAT(' OUTPUT STATISTICS')
      WRITE(6,1001)(STAT(I),I=1,4)
      WRITE(6,982)
  982 FORMAT(' NUMBER OF MISSING VALUES')
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      WRITE(6,1010)NMISS
C
C CALCULATE WINDOW WIDTH, HH
C
      HH=(BNDS(2)-BNDS(1))/(NODE-1)
C
C CALCULATE VALUES OF LOG OF CURRENT CYCLES AT WHICH PDF IS ESTIMATED;
C ALSO CALLED 'NODE' VALUES
C
      DO 6001,I=1,NODE-2
      BNDS(I+2)=BNDS(1) + (I*HH)
 6001 CONTINUE
      WRITE(6,983)
  983 FORMAT(' LOG OF CURRENT CYCLES, LOG XNF')
      WRITE(6,1001)(BNDS(I),I=1,NODE)
C
C REORDER BNDS FOR PLOTTING
C
      SAVE1 = BNDS(2)
      SAVE2 = BNDS(NODE)
      BNDS(NODE)=BNDS(2)
      DO 6002, I=1, NODE-2
      BNDS(I+1)=BNDS(I+2)
 6002 CONTINUE
      BNDS(NODE-1)=SAVE2
      BNDS(NODE)=SAVE1
      WRITE(6,984)
  984 FORMAT(' ORDERED LOG OF CURRENT CYCLES, LOG,XNF.
     1X AXIS PDF, CDF PLOT')
      WRITE(6,1001)(BNDS(I),I=1,NODE)
CC WRITE LOG OF CURRENT CYCLES AND PDF OF LOG OF CURRENT CYCLES,
CC   LOG XNF TO PLOT FILES
      WRITE(34,990)
  990 FORMAT(' (E12.4,1X,E12.4)')
      WRITE(34,991)(BNDS(J),DENS(J),J=1,NODE)
  991 FORMAT(E12.4,1X,E12.4)
C
C CALCULATE CDF OF LOG OF CURRENT CYCLES
C
      READ(5,1010)IOPT
      WRITE(6,992)
  992 FORMAT(' GCDF PARAMETERS')
      WRITE(6,1010)IOPT
      XO=BNDS(1)
      DO 6003,I=1,NODE
      P=GCDF(XO,IOPT,NODE,BNDS,DENS)
      BNDSX(I)=XO
      XO=XO+HH
      DISTX(I)=P
 6003 CONTINUE
      WRITE(6,994)
  994 FORMAT(' CDF OF LOG OF CURRENT CYCLES, LOG XNF,
     1Y AXIS OF PDF, CDF PLOT')
      WRITE(6,1001)(DISTX(I),I=1,NODE)
C
      WRITE(6,993)
  993 FORMAT(' ORDERED LOG OF CURRENT CYCLES, LOG XNF,
     1X AXIS OF PDF, CDF PLOT')
      WRITE(6,1001)(BNDS(I),I=1,NODE)
      WRITE(6,1001)(BNDSX(I),I=1,NODE)
C
C WRITE LOG OF CURRENT CYCLES AND CDF OF LOG OF CURRENT
C TO THE PLOT FILES
C
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      WRITE(35,990)
      WRITE(35,991)(BNDS(J),DISTX(J),J=1,NODE)
      STOP
      END
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
      SUBROUTINE SORT (Y,N)
      DIMENSION Y(10000)
C Y IS THE ARRAY TO BE SORTED
C AT COMPLETION Y(1) IS SMALLEST VALUE
C AT COMPLETION Y(N) IS LARGEST VALUE
      N1 = N - 1
      DO 1 I=1,N1
      J = I + 1
      DO 2 K=J,N
      IF (Y(I).LT.Y(K))GO TO 2
      TEMP = Y(I)
      Y(I) = Y(K)
      Y(K) = TEMP
    2 CONTINUE
    1 CONTINUE
      RETURN
      END
C-----------------------------------------------------------------------
C  IMSL Name:  D3SPL/DD3SPL (Single/Double precision version)
C
C  Computer:   IBM/SINGLE
C
C  Revised:    November 1, 1985
C
C  Purpose:    Nonparametric probability density function estimation
C              estimation by the penalized likelihood method.
C
C  Usage:      CALL D3SPL (NOBS, X, NODE, BNDS, INIT, ALPHA, MAXIT, EPS,
C                          DENS, STAT, HESS, LDHESS, ILOHI, DENEST, B,
C                          IPVT, WK2)
C
C  Arguments:
C     NOBS   - Number of observations.  (Input)
C     X      - Vector of length NOBS containing the random sample of
C              responses.  (Input)
C     NODE   - Number of mesh nodes for the discrete pdf estimate.
C              (Input)
C     BNDS   - Vector of length 2 containing the minimum and maximum
C              values for X(i) in BNDS(1) and BNDS(2), respectively.
C              (Input)
C     INIT   - Initialization option.  (Input)
C     ALPHA  - Positive penalty weighting factor which controls the
C              smoothness of the estimate.  (Input)
C     MAXIT  - Maximum number of iterations allowed in the iterative
C              procedure.  (Input)
C     EPS    - Convergence criterion.  (Input)
C     DENS   - Vector of length NODE containing the estimated values of
C              the discrete pdf at the NODE equally spaced mesh nodes.
C              (Input/output if INIT=1, Output otherwise)
C     STAT   - Vector of length 4 containing out statistics.  (Output)
C              STAT(1) and STAT(2) contain the log-likelihood and the
C              log-penalty terms, respectively.  STAT(3) and STAT(4)
C              contain the estimated mean and variance for the
C              estimated density.
C     HESS   - Seven by NODE-2 hessian matrix (and its factorization).
C              (Output)
C     LDHESS - Leading dimension of HESS exactly as specified in the
C              dimension statement in the calling program.  (Input)
C     ILOHI  - NODE by 2 matrix containing the indices for the risk set
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C                at each node value.  (Output)
C     DENEST  - NODE by 3matrix containing the gradient vector, among
C                other quantities.  (Output)
C     B       - Vector of length NODE containing the NODE values.
C                (Output)
C     IPVT    - Pivot vector of length NODE-2.  (Output)
C     WK2     - Work vector of length NODE-2.  (Output)
C
C   Chapter:    STAT/LIBRARY Density and Hazard Estimation
C
C   Copyright:  1985 by IMSL, Inc.  All Rights Reserved.
C
C   Warranty:   IMSL warrants only that IMSL testing has been applied
C               to this code.  No other warranty, expressed or implied,
C               is applicable.
C
C-----------------------------------------------------------------------
C
      SUBROUTINE D3SPL (NOBS, X, NODE, BNDS, INIT, ALPHA, MAXIT, EPS,
     &                  DENS, STAT, HESS, LDHESS, ILOHI, DENEST, B,
     &                  IPVT, WK2)
C                                  SPECIFICATIONS FOR ARGUMENTS
      INTEGER    NOBS, NODE, INIT, MAXIT, LDHESS, ILOHI(NODE,*),
     &           IPVT(*)
      REAL       ALPHA, EPS, X(*), BNDS(2), DENS(*), STAT(4),
     &           HESS(LDHESS,*), DENEST(NODE,*), B(*), WK2(*)
C                                  SPECIFICATIONS FOR LOCAL VARIABLES
      INTEGER    I, IMPTR, IPTR, ITER, K, KM1, KM2, KP1, KP2, M, MOLD,
     &           NER, NOB1
      REAL       BK, BKM1, BSMALL, CK, CKM1, CKM2, CKMCM1, CKP1, CKP2,
     &           CONS, EPS1, FACTOR, FK, FKM1, FKM2, FKP1, H, H2, H3,
     &           SUM, TEMP, WK(4)
      DOUBLE PRECISION SUM1, SUM2, SUM3
C                                  SPECIFICATIONS FOR SAVE VARIABLES
      INTEGER    MINCR(8)
      SAVE       MINCR
C                                  SPECIFICATIONS FOR INTRINSICS
C     intrinsic  alog,amax1,max0,min0,mod,sqrt
      INTRINSIC  ALOG, AMAX1, MAX0, MIN0, MOD, SQRT
      INTEGER    MAX0, MIN0, MOD
      REAL       ALOG, AMAX1, SQRT
C                                  SPECIFICATIONS FOR SUBROUTINES
      EXTERNAL   E1MES, E1POP, E1PSH, E1STI, E1STR, SADD, SAXPY,
     &           SCOPY, SHPROD, SSCAL, D2SPT, L2TRB, LFSRB
C                                  SPECIFICATIONS FOR FUNCTIONS
      EXTERNAL   ISMIN, N1RCD, SDOT, SNRM2, SSUM
      INTEGER    ISMIN, N1RCD
      REAL       SDOT, SNRM2, SSUM
C
      DATA MINCR/5, 9, 17, 33, 65, 129, 253, 100001/
C
      CALL E1PSH ('D3SPL ')
C                                  Error checks
      NER = 1
      IF (NOBS .LT. 1) THEN
         CALL E1MES (5, 1, 'After removing all missing (NaN, not a '//
     &               'number) values from X there are no valid '//
     &               'observations.  At least one valid observation '//
     &               'is necessary.')
      END IF
      IF (NODE .LE. 4) THEN
         CALL E1STI (1, NODE)
         CALL E1MES (5, 2, 'NODE = %(I1).  The number of mesh '//
     &               'nodes, NODE, must be an odd integer greater '//
```

ORIGINAL PAGE IS OF POOR QUALITY

```
     &                    'than 4.')
      ELSE IF (MOD(NODE,2) .EQ. 0) THEN
         CALL E1STI (1, NODE)
         CALL E1MES (5, 3, 'NODE = %(I1) must be an odd integer '//
     &                    'greater than 4.')
      END IF
      IF (ALPHA .LE. 0.0) THEN
         CALL E1STR (1, ALPHA)
         CALL E1MES (5, 4, 'ALPHA = %(R1).  The penalty weighting '//
     &               'factor which controls smoothness, ALPHA, must '//
     &               'be greater than 0.')
      END IF
      IF (MAXIT .LE. 0.0) THEN
         CALL E1STI (1, MAXIT)
         CALL E1MES (5, 5, 'MAXIT = %(I1).  The maximum number '//
     &               'of iterations, MAXIT, must be greater than 0.')
      END IF
      IF (BNDS(1) .GT. BNDS(2)) THEN
         CALL E1STR (1, BNDS(1))
         CALL E1STR (2, BNDS(2))
         CALL E1MES (5, 6, 'BNDS(1) = %(R1) and BNDS(2) = '//
     &               '%(R2).  The minimum value for X, BNDS(1), must '//
     &               'be less than or equal to the maximum value for '//
     &               'X, BNDS(2).')
      END IF
      IF (INIT .NE. 0) THEN
         IF (DENS(1).NE.0 .OR. DENS(NODE).NE.0) THEN
            CALL E1STR (1, DENS(1))
            CALL E1STR (2, DENS(NODE))
            CALL E1STI (1, NODE)
            CALL E1MES (5, 7, 'DENS(1) = %(R1) and DENS(NODE=%(I1)) '//
     &                  '= %(R2).  The beginning and ending initial '//
     &                  'estimates of the density must be zero.')
         END IF
         IF (DENS(ISMIN(NODE,DENS,1)) .LT. 0) THEN
            CALL E1MES (5, 8, 'The initial estimates of the '//
     &                  'density, DENS, must be greater than or '//
     &                  'equal to 0.')
         END IF
      END IF
      NOB1 = 0
      DO 10  I=1, NOBS
         IF (X(I).LT.BNDS(1) .OR. X(I).GT.BNDS(2)) THEN
            NOB1 = NOB1 + 1
         END IF
   10 CONTINUE
      IF (NOB1 .EQ. NOBS) THEN
         CALL E1MES (5, 9, 'All elements in X lie outside the '//
     &                'interval BNDS(1) to BNDS(2).  At least one '//
     &                'element of X must lie in this interval.')
      END IF
      IF (EPS .LE. 0.0) THEN
         EPS1 = 1.0E-4
      ELSE
         EPS1 = EPS
      END IF
      IF (N1RCD(0) .NE. 0) GO TO 9000
C                                  Initialization
      IMPTR = 0
C                                  Set initial densities
      IF (INIT .EQ. 0) THEN
         DENS(1) = 0.0
         DENS(2) = 2.0/(BNDS(2)-BNDS(1))
         DENS(3) = 0.0
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
         M = 3
      ELSE
         M = NODE
      END IF
C                                       Refine mesh
   20 IF (INIT .EQ. 0) THEN
         MOLD = M
         IMPTR = IMPTR + 1
         M = MINO(NODE,MINCR(IMPTR))
      END IF
C                                       Get mesh interval width
      H = (BNDS(2)-BNDS(1))/(M-1)
      H2 = H*H
      H3 = H2*H
C                                       Make initial DENS integrate to 1.
      IF (INIT .NE. 0) THEN
         CALL SSCAL (NODE, 1.0/(H*SSUM(NODE,DENS,1)), DENS, 1)
      END IF
C                                       Set mesh nodes
      B(1) = BNDS(1)
      DO 30  I=2, M
         B(I) = B(I-1) + H
   30 CONTINUE
C                                       Set B indices for interpolating X
      IPTR = 0
   40 IPTR = IPTR + 1
      IF (X(IPTR) .LT. BNDS(1)) GO TO 40
      DO 60  K=1, M - 1
         ILOHI(K,1) = IPTR
         ILOHI(K,2) = IPTR - 1
         IF (IPTR .LE. NOBS) THEN
   50       IF (X(IPTR) .LT. B(K+1)) THEN
               ILOHI(K,2) = ILOHI(K,2) + 1
               IPTR = IPTR + 1
               IF (IPTR .LE. NOBS) GO TO 50
            END IF
         END IF
   60 CONTINUE
   70 FACTOR = 2.0*ALPHA/H3
C                                       Initialize mesh node densities
      IF (INIT .EQ. 0) THEN
C                                       Via DESPT
         CALL D2SPT (M-2, B(2), 1, MOLD, BNDS, DENS, DENEST, WK, WK,
     &               WK)
         TEMP = 1.0/(M*M*H)
         DO 80  I=2, M - 1
            DENS(I) = AMAX1(TEMP,SQRT(DENEST(I-1,1)))
   80    CONTINUE
      ELSE
C                                       Via the initial estimates
         DO 90  I=2, M - 1
            DENS(I) = SQRT(DENS(I))
   90    CONTINUE
      END IF
      DENS(M) = 0.0
C                                       Maximize
      DO 140  ITER=1, MAXIT
C                                       Get Hessian - Lagrangian
         HESS(1,1) = 0.0
         HESS(1,2) = 0.0
         HESS(2,1) = 0.0
         BSMALL = 0.0
         SUM = 0.0
C                                       CK** are true estimates = FK**2
```

```
      DO 120  K=2, M - 1
         KM1 = K - 1
         KM2 = MAX0(1,K-2)
         KP1 = K + 1
         KP2 = MINO(M,K+2)
         FK = DENS(K)
         FKM1 = DENS(KM1)
         FKM2 = DENS(KM2)
         CKM2 = FKM2**2
         CKM1 = FKM1**2
         CK = FK**2
         CKP1 = DENS(KP1)**2
         CKP2 = DENS(KP2)**2
         BK = B(K)
         BKM1 = B(KM1)
         SUM = SUM + CK
         IF (K .GE. 4) HESS(1,KM1) = 4.0*FK*FKM2*FACTOR
         SUM1 = 0.0D0
         SUM2 = 0.0D0
         SUM3 = 0.0D0
         DO 100  I=ILOHI(K,1), ILOHI(K,2)
            TEMP = (X(I)-BK)/H
            CONS = (1.0-TEMP)/(CK+(CKP1-CK)*TEMP)
            SUM1 = SUM1 - CONS
            SUM2 = SUM2 + CONS*CONS
  100    CONTINUE
         CKMCM1 = CK - CKM1
         DO 110  I=ILOHI(KM1,1), ILOHI(KM1,2)
            CONS = (X(I)-BKM1)/H
            TEMP = CKM1 + CKMCM1*CONS
            SUM1 = SUM1 - CONS/TEMP
            TEMP = TEMP*TEMP
            SUM2 = SUM2 + (CONS*CONS)/TEMP
            SUM3 = SUM3 + CONS*(1.0-CONS)/TEMP
  110    CONTINUE
         TEMP = FACTOR*(CKM2+CKP2-4.0*(CKM1+CKP1)+6.0*CK) + SUM1
         TEMP = 2.0*TEMP
         BSMALL = BSMALL + 2.0*CK*TEMP
         HESS(3,KM1) = TEMP + 4.0*CK*(6.0*FACTOR+SUM2)
         IF (K .NE. 2) HESS(2,KM1) = 4.0*FK*FKM1*(-4.0*FACTOR+SUM3)
         DENEST(KM1,1) = FK*TEMP
         DENEST(KM1,2) = -2.0*FK
  120 CONTINUE
      BSMALL = 1.0/H - SUM + BSMALL
C                                  Save portion of DENEST
      CALL SCOPY (M-2, DENEST(1,2), 1, DENEST(1,3), 1)
C                                  Finish with the hessian
      CALL SADD (M-2, -BSMALL/(2.0*SUM), HESS(3,1), LDHESS)
C                                  Fill out symmetric band structure
      CALL SCOPY (M-4, HESS(1,3), LDHESS, HESS(5,1), LDHESS)
      HESS(5,M-3) = 0.0
      HESS(5,M-2) = 0.0
      CALL SCOPY (M-3, HESS(2,2), LDHESS, HESS(4,1), LDHESS)
      HESS(4,M-2) = 0.0
C                                  Solve symmetric band linear system
      CALL L2TRB (M-2, HESS, LDHESS, 2, 2, HESS, LDHESS, IPVT, WK2)
      CALL LFSRB (M-2, HESS, LDHESS, 2, 2, IPVT, DENEST, 1, DENEST)
      CALL LFSRB (M-2, HESS, LDHESS, 2, 2, IPVT, DENEST(1,2), 1,
     &            DENEST(1,2))
      IF (N1RCD(1) .NE. 0) GO TO 9000
C                                  Compute the constant
      CONS = SDOT(M-2,DENEST(1,3),1,DENEST(1,2),1)
      CONS = (1.0/H-SUM-SDOT(M-2,DENEST(1,3),1,DENEST(1,1),1))/CONS
C                                  Update the gradient
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
         CALL SAXPY (M-2, CONS, DENEST(1,2), 1, DENEST(1,1), 1)
C                                       Parameter updates
         CALL SAXPY (M-2, -1.0, DENEST(1,1), 1, DENS(2), 1)
C                                       Check the convergence criterion
         TEMP = SNRM2(M-2,DENS(2),1)
         IF (SNRM2(M-2,DENEST,1) .LT. EPS1*TEMP) GO TO 150
C                                       Ad hoc projection to plus quadrant
         TEMP = TEMP*1.0E-4/SQRT(M-2.0)
         DO 130  I=2, M - 1
            DENS(I) = AMAX1(TEMP,DENS(I))
  130    CONTINUE
  140 CONTINUE
      CALL E1STI (1, MAXIT)
      CALL E1MES (3, 1, 'The maximum number of iterations '//
     &            '(MAXIT=%(I1)) was exceeded.')
C                                       Replace DENS(*) with squares
  150 CALL SHPROD (M-2, DENS(2), 1, DENS(2), 1, DENS(2), 1)
      IF (M .NE. NODE) GO TO 20
C                                       Evaluate log likelihood and penalty
      SUM1 = 0.0
C                                       Penalty
      DO 160  K=1, M
         KM1 = MAX0(K-1,1)
         KP1 = MIN0(K+1,M)
         SUM1 = SUM1 + (DENS(KM1)-2.0*DENS(K)+DENS(KP1))**2
  160 CONTINUE
      STAT(2) = -0.5*FACTOR*SUM1
      SUM2 = 0.0
C                                       Log-likelihood
      DO 170  I=1, NOBS
         IF (X(I).GE.BNDS(1) .AND. X(I).LE.BNDS(2)) THEN
            CALL D2SPT (1, X(I), 1, NODE, BNDS, DENS, DENEST, WK, WK,
     &                  WK)
            SUM2 = SUM2 + ALOG(DENEST(1,1))
         END IF
  170 CONTINUE
      STAT(1) = SUM2
C                                       Evaluate M.L.P.E. mean and variance
      SUM1 = 0.0
      SUM2 = 0.0
      DO 180  K=1, M - 1
         FK = DENS(K)
         FKP1 = DENS(K+1)
         BK = B(K)
         CONS = FK + FKP1
         TEMP = CONS + FKP1
         SUM1 = SUM1 + H2*TEMP/6.0 + 0.5*H*BK*CONS
         SUM2 = SUM2 + H3*(TEMP+FKP1)/12.0 + H2*BK*TEMP/3.0 +
     &          0.5*H*BK*BK*CONS
  180 CONTINUE
      STAT(3) = SUM1
      STAT(4) = SUM2 - SUM1*SUM1
C                                       Exit section
 9000 CALL E1POP ('D3SPL ')
      RETURN
      END
/EOF
```

```
EEEEEEEEEE   4444444444444444444444444444444444444444444444444444444444444444444444444444444444444444   EEEEEEEE
EEEEEEEEEE   44444444444444444444444444  Digital Equipment Corporation - VAX/VMS Version V4.7  4444444444444444444444444   EEEEEEEE
EEEEEEEEEE   4444444444444444444444444444444444444444444444444444444444444444444444444444444444444444   EEEEEEEE
```

DECNET

NORMAL

INP;12

File _DUA0:[]NORMAL.INP;12 (487,157,0), last revised on 29-NOV-1988 11:22, is a 1 block sequential file owned by UIC [DECNET]. The records are variable length with implied (CR) carriage control. The longest record is 39 bytes.

Job NORMAL (1815) queued to TERM$LA120A on 29-NOV-1988 11:22 by user DECNET, UIC [DECNET], under account DECNET at priority 100, started on printer LTA4: on 29-NOV-1988 11:24 from queue TERM$LA120A.

```
EEEEEEEEEE   4444444444444444444444444444444444444444444444444444444444444444444444444444444444444444   EEEEEEEE
EEEEEEEEEE   44444444444444444444444444  Digital Equipment Corporation - VAX/VMS Version V4.7  4444444444444444444444444   EEEEEEEE
EEEEEEEEEE   4444444444444444444444444444444444444444444444444444444444444444444444444444444444444444   EEEEEEEE
```

	1	40		
	28.0E-01	1.4E-01		
	300.0E-06	45.0E-06		
	2.20E-11	0.22E-11		
	6.2E+02	6.2E+01		
21 2	0	50.0E-01	10.0E-05	30

ORIGINAL PAGE IS
OF POOR QUALITY

```
CCCCCCCCCC   2222222222222222222222222222222222222222222222222222222222222222222222222222222222222222   CCCCCCCC
CCCCCCCCCC   22222222222222222222222222  Digital Equipment Corporation - VAX/VMS Version V4.7  222222222222222222222222222   CCCCCCCC
CCCCCCCCCC   2222222222222222222222222222222222222222222222222222222222222222222222222222222222222222   CCCCCCCC
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
DECNET

RANDM22

CPR;1
```


File _DUA0:[]RANDM22.CPR;1 (487,156,0), last revised on 29-NOV-1988 11:19, is a 24 block sequential file owned by UIC [DECNET]. The records are variable length with FORTRAN (FTN) carriage control. The longest record is 120 bytes.

Job RANDM22 (1813) queued to TERM$LA120A on 29-NOV-1988 11:19 by user DECNET, UIC [DECNET], under account DECNET at priority 100, started on printer LTA4: on 29-NOV-1988 11:19 from queue TERM$LA120A.


```
CCCCCCCCCC   2222222222222222222222222222222222222222222222222222222222222222222222222222222222222222   CCCCCCCC
CCCCCCCCCC   22222222222222222222222222  Digital Equipment Corporation - VAX/VMS Version V4.7  222222222222222222222222222   CCCCCCCC
CCCCCCCCCC   2222222222222222222222222222222222222222222222222222222222222222222222222222222222222222   CCCCCCCC
```

```
        1             40
  0.2800E+01   0.1400E+00
 MATERIAL PROPERTY, M
  0.2195E+01   0.2643E+01   0.2897E+01   0.2785E+01   0.2812E+01
  0.2691E+01   0.2566E+01   0.2865E+01   0.2865E+01   0.3012E+01
  0.2759E+01   0.2807E+01   0.2934E+01   0.2546E+01   0.2574E+01
  0.2810E+01   0.2862E+01   0.2461E+01   0.2758E+01   0.2590E+01
  0.2771E+01   0.2868E+01   0.2831E+01   0.3007E+01   0.2943E+01
  0.2809E+01   0.2614E+01   0.2855E+01   0.2770E+01   0.2874E+01
  0.2988E+01   0.2900E+01   0.2711E+01   0.2566E+01   0.2888E+01
  0.2738E+01   0.2847E+01   0.2897E+01   0.3131E+01   0.2752E+01
        1             40
  0.3000E-03   0.4500E-04
 INITIAL CRACK SIZE, AI
  0.2226E-03   0.3291E-03   0.2797E-03   0.2715E-03   0.3743E-03
  0.2921E-03   0.2942E-03   0.2591E-03   0.2747E-03   0.2515E-03
  0.3127E-03   0.3614E-03   0.3482E-03   0.2739E-03   0.3140E-03
  0.2827E-03   0.2768E-03   0.2971E-03   0.3925E-03   0.2652E-03
  0.4206E-03   0.2786E-03   0.3617E-03   0.2727E-03   0.3325E-03
  0.2552E-03   0.2576E-03   0.3440E-03   0.3047E-03   0.3328E-03
  0.2800E-03   0.3017E-03   0.2586E-03   0.2697E-03   0.2558E-03
  0.3213E-03   0.3738E-03   0.2246E-03   0.3674E-03   0.3441E-03
        1             40
  0.2200E-10   0.2200E-11
 MATERIAL PROPERTY, C
  0.1807E-10   0.2346E-10   0.2104E-10   0.2063E-10   0.2557E-10
  0.2167E-10   0.2177E-10   0.1999E-10   0.2079E-10   0.1960E-10
  0.2267E-10   0.2498E-10   0.2436E-10   0.2075E-10   0.2274E-10
  0.2120E-10   0.2090E-10   0.2191E-10   0.2640E-10   0.2031E-10
  0.2765E-10   0.2099E-10   0.2499E-10   0.2069E-10   0.2362E-10
  0.1979E-10   0.1992E-10   0.2417E-10   0.2229E-10   0.2364E-10
  0.2106E-10   0.2214E-10   0.1997E-10   0.2054E-10   0.1983E-10
  0.2309E-10   0.2555E-10   0.1817E-10   0.2526E-10   0.2417E-10
        1             40
  0.6200E+03   0.6200E+02
 STRESS RANGE, DELSIG
  0.5092E+03   0.6612E+03   0.5931E+03   0.5814E+03   0.7207E+03
  0.6106E+03   0.6135E+03   0.5635E+03   0.5859E+03   0.5524E+03
  0.6390E+03   0.7039E+03   0.6866E+03   0.5848E+03   0.6408E+03
  0.5973E+03   0.5889E+03   0.6175E+03   0.7439E+03   0.5723E+03
  0.7791E+03   0.5915E+03   0.7043E+03   0.5831E+03   0.6658E+03
  0.5578E+03   0.5613E+03   0.6811E+03   0.6281E+03   0.6662E+03
  0.5935E+03   0.6238E+03   0.5627E+03   0.5788E+03   0.5587E+03
  0.6507E+03   0.7200E+03   0.5121E+03   0.7118E+03   0.6812E+03
 LOG OF CYCLES TO REACH CRACK SIZE=2.0E-03M,
 IVEN STRESS MEAN AMPLITUDE=6.2E+02MPA
  0.4907E+01   0.3774E+01   0.3633E+01   0.3834E+01   0.3334E+01
  0.3867E+01   0.4041E+01   0.3789E+01   0.3705E+01   0.3630E+01
  0.3674E+01   0.3393E+01   0.3253E+01   0.4164E+01   0.3941E+01
  0.3741E+01   0.3698E+01   0.4182E+01   0.3348E+01   0.4144E+01
  0.3228E+01   0.3680E+01   0.3354E+01   0.3514E+01   0.3309E+01
  0.3889E+01   0.4149E+01   0.3390E+01   0.3693E+01   0.3411E+01
  0.3502E+01   0.3518E+01   0.4007E+01   0.4155E+01   0.3775E+01
  0.3666E+01   0.3280E+01   0.3952E+01   0.2869E+01   0.3548E+01
 SORTED LOG OF CYCLES
  0.2868E+01   0.3228E+01   0.3253E+01   0.3280E+01   0.3309E+01
  0.3334E+01   0.3348E+01   0.3354E+01   0.3390E+01   0.3393E+01
  0.3411E+01   0.3502E+01   0.3514E+01   0.3518E+01   0.3548E+01
  0.3630E+01   0.3633E+01   0.3666E+01   0.3674E+01   0.3680E+01
  0.3693E+01   0.3698E+01   0.3705E+01   0.3741E+01   0.3774E+01
  0.3775E+01   0.3789E+01   0.3834E+01   0.3867E+01   0.3889E+01
  0.3941E+01   0.3952E+01   0.4007E+01   0.4041E+01   0.4144E+01
  0.4149E+01   0.4155E+01   0.4164E+01   0.4182E+01   0.4907E+01
 DESPL PARAMETERS
```

```
21    0    0.5000E+01  0.1000E-03    30
NDS(1),BNDS(2)=   0.2724E+01    0.5153E+01
DF OF LOG OF CURRENT CYCLES,LOG XNF,Y AXIS OF PDF PLOT
  0.0000E+00  0.5049E-01  0.1355E+00  0.2417E+00  0.3572E+00
  0.4706E+00  0.5719E+00  0.6542E+00  0.7124E+00  0.7429E+00
  0.7450E+00  0.7193E+00  0.6684E+00  0.5957E+00  0.5065E+00
  0.4067E+00  0.3027E+00  0.2019E+00  0.1118E+00  0.4101E-01
  0.0000E+00
OUTPUT STATISTICS
 -0.2267E+02 -0.3434E+02  0.3913E+01  0.2276E+00
NUMBER OF MISSING VALUES
  0
LOG OF CURRENT CYCLES, LOG XNF
  0.2724E+01  0.5153E+01  0.2846E+01  0.2967E+01  0.3089E+01
  0.3210E+01  0.3332E+01  0.3453E+01  0.3574E+01  0.3696E+01
  0.3817E+01  0.3939E+01  0.4060E+01  0.4182E+01  0.4303E+01
  0.4424E+01  0.4546E+01  0.4667E+01  0.4789E+01  0.4910E+01
  0.5031E+01
ORDERED LOG OF CURRENT CYCLES, LOG,XNF,                 X AXIS PDF, CDF PLOT
  0.2724E+01  0.2846E+01  0.2967E+01  0.3089E+01  0.3210E+01
  0.3332E+01  0.3453E+01  0.3574E+01  0.3696E+01  0.3817E+01
  0.3939E+01  0.4060E+01  0.4182E+01  0.4303E+01  0.4424E+01
  0.4546E+01  0.4667E+01  0.4789E+01  0.4910E+01  0.5031E+01
  0.5153E+01
GCDF PARAMETERS
  2
CDF OF LOG OF CURRENT CYCLES, LOG XNF,                  Y AXIS OF PDF, CDF PLOT
  0.0000E+00  0.3065E-02  0.1435E-01  0.3725E-01  0.7362E-01
  0.1239E+00  0.1872E+00  0.2616E+00  0.3446E+00  0.4329E+00
  0.5232E+00  0.6121E+00  0.6964E+00  0.7731E+00  0.8400E+00
  0.8955E+00  0.9385E+00  0.9692E+00  0.9882E+00  0.9975E+00
  0.1000E+01
ORDERED LOG OF CURRENT CYCLES, LOG XNF,                 X AXIS OF PDF, CDF PLOT
  0.2724E+01  0.2846E+01  0.2967E+01  0.3089E+01  0.3210E+01
  0.3332E+01  0.3453E+01  0.3574E+01  0.3696E+01  0.3817E+01
  0.3939E+01  0.4060E+01  0.4182E+01  0.4303E+01  0.4424E+01
  0.4546E+01  0.4667E+01  0.4789E+01  0.4910E+01  0.5031E+01
  0.5153E+01
  0.2724E+01  0.2846E+01  0.2967E+01  0.3089E+01  0.3210E+01
  0.3332E+01  0.3453E+01  0.3574E+01  0.3696E+01  0.3817E+01
  0.3939E+01  0.4060E+01  0.4182E+01  0.4303E+01  0.4424E+01
  0.4546E+01  0.4667E+01  0.4789E+01  0.4910E+01  0.5031E+01
  0.5153E+01
```

ORIGINAL PAGE IS OF POOR QUALITY

```
11:17:04.4303   0.0000  CSP          University of Texas System Center for High Performance Computing
11:17:04.4306   0.0000  CSP
11:17:04.4308   0.0000  CSP          CRAY X-MP/24  S/N-130/89/26      Austin, Texas             11/29/88
11:17:04.4311   0.0000  CSP
11:17:04.4314   0.0000  CSP          CRAY Operating System            COS 1.16 BF2  Assembled 10/24/88
11:17:04.4317   0.0000  CSP
11:17:04.4319   0.0000  CSP
11:17:04.8613   0.0000  CSP     JOB,JN=RANDM22,US=USAQ530,RT=60,T=30,MFL=3000000.
11:17:04.8856   0.0008  CSP     ACCOUNT,UPW=.
11:17:07.2151   0.1023  CSP     lib,dn=imslib.
11:17:11.0050   0.1357  PDM     PD000 - PDN = IMSLIB            ID = STATIC     ED =     4  OWN = LIBRARY
11:17:11.0055   0.1357  PDM     PD000 - ACCESS  COMPLETE
11:17:11.0311   0.1373  CSP     FETCH,DN=AAA,TEXT=.
11:17:12.3536   0.1374  SCP      VAX TO CRAY: %SYSTEM-S-NORMAL, normal successful completion
11:17:12.3538   0.1374  SCP      VAX TO CRAY: FILE=$1$DUA2:[USAQ530.RANDOM2.NEWRAND2]NORMAL.INP;12
11:17:12.3541   0.1374  SCP      VAX TO CRAY: 248 BYTES TRANSFERRED
11:17:12.9106   0.1374  SCP     SS004 - DATASET RECEIVED FROM FRONT END
11:17:12.9672   0.1374  CSP     ASSIGN,DN=AAA,A=FT05.
11:17:12.9695   0.1376  CSP     ASSIGN,DN=RRR,A=FT34.
11:17:12.9722   0.1378  CSP     ASSIGN,DN=SSS,A=FT35.
11:17:12.9749   0.1382  CSP     ACCESS,DN=$BLD,PDN=NR2BLD,ID=SMBOYCE.
11:17:13.0130   0.1383  PDM     PD000 - PDN = NR2BLD            ID = SMBOYCE    ED =     3  OWN = USAQ530
11:17:13.0132   0.1383  PDM     PD000 - ACCESS  COMPLETE
11:17:13.0375   0.1386  CSP     SEGLDR,CMD='LIB=IMSLIB;DUPENTRY=IGNORE',GO.
11:17:13.1083   0.1393  USER    SG000 - SEGLDR VERSION 3.1  - 05/24/88     (CHG-12/31/86)
11:18:01.3209   7.6090  USER    SG001 - BEGIN EXECUTION
11:18:01.4544   7.6735  USER    UT010 - STOP           in $MAIN
11:18:01.4660   7.6737  CSP     DISPOSE,DN=RRR,SDN=OUT1.
11:18:01.5833   7.6739  CSP     DISPOSE,DN=SSS,SDN=OUT2.
11:18:01.7168   7.6740  CSP     END OF JOB
11:18:02.9591   7.6740  SCP      CRAY TO VAX: %RMS-S-NORMAL, normal successful completion
11:18:02.9594   7.6740  SCP      CRAY TO VAX: FILE=$1$DUA2:[USAQ530.RANDOM2.NEWRAND2]OUT1.CPR;1
11:18:02.9596   7.6740  SCP      CRAY TO VAX: 542 BYTES TRANSFERRED
11:18:02.9819   7.6740  SCP      CRAY TO VAX: %RMS-S-NORMAL, normal successful completion
11:18:02.9822   7.6740  SCP      CRAY TO VAX: FILE=$1$DUA2:[USAQ530.RANDOM2.NEWRAND2]OUT2.CPR;1
11:18:02.9825   7.6740  SCP      CRAY TO VAX: 542 BYTES TRANSFERRED
11:18:03.0361   7.6740  CSP
11:18:03.0363   7.6740  CSP
11:18:03.1065   7.6742  USER        JOB NAME -                              RANDM22
11:18:03.1068   7.6742  USER        USER NUMBER -                           USAQ530
11:18:03.1071   7.6742  USER        JOB SEQUENCE NUMBER -                      4788
11:18:03.1074   7.6742  USER
11:18:03.1077   7.6742  USER        TIME EXECUTING IN CPU -                 0000:00:07.6742
11:18:03.1080   7.6742  USER        TIME WAITING TO EXECUTE -               0000:00:41.9199
11:18:03.1083   7.6742  USER        TIME WAITING FOR I/O -                  0000:00:08.2839
11:18:03.1086   7.6742  USER        TIME WAITING SEMAPHORE -                0000:00:00.0000
11:18:03.1098   7.6743  USER        TIME WAITING IN INPUT QUEUE -           0000:00:00.0765
11:18:03.1101   7.6743  USER        MEMORY * CPU TIME (MWDS*SEC) -                 1.79962
11:18:03.1104   7.6743  USER        MEMORY * I/O WAIT TIME (MWDS*SEC) -            1.81682
11:18:03.1107   7.6743  USER        MEMORY * SEM WAIT TIME (MWDS*SEC) -            0.00000
11:18:03.1110   7.6743  USER        MINIMUM JOB SIZE (WORDS) -                52736
11:18:03.1115   7.6743  USER        MAXIMUM JOB SIZE (WORDS) -               472064
11:18:03.1119   7.6743  USER        MINIMUM FL (WORDS) -                      47104
11:18:03.1122   7.6744  USER        MAXIMUM FL (WORDS) -                     466432
11:18:03.1125   7.6744  USER        MINIMUM JTA (WORDS) -                      4096
11:18:03.1128   7.6744  USER        MAXIMUM JTA (WORDS) -                      5632
11:18:03.1131   7.6744  USER        DISK SECTORS MOVED -                       6194
11:18:03.1134   7.6744  USER        FSS SECTORS MOVED -                           0
11:18:03.1137   7.6744  USER        USER I/O REQUESTS -                         467
11:18:03.1140   7.6744  USER        USER I/O SUSPENSIONS -                     1358
11:18:03.1143   7.6744  USER        OPEN CALLS -                                 26
11:18:03.1146   7.6744  USER        CLOSE CALLS -                                28
11:18:03.1149   7.6744  USER        MEMORY RESIDENT DATASETS -                    0
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
11:18:03.1152   7.6744  USER   TEMPORARY DATASET SECTORS USED -          5
11:18:03.1155   7.6744  USER   PERMANENT DATASET SECTORS ACCESSED -   4792
11:18:03.1158   7.6745  USER   PERMANENT DATASET SECTORS SAVED -         0
11:18:03.1161   7.6745  USER   SECTORS RECEIVED FROM FRONT END -         1
11:18:03.1164   7.6745  USER   SECTORS QUEUED TO FRONT END -             2
11:18:03.1167   7.6745  USER
11:18:03.1170   7.6745  USER       7.6742 CPU seconds         =     5.7557 RT seconds ( 82%)
11:18:03.1173   7.6745  USER          467 I/O requests        =     0.1401 RT seconds (  2%)
11:18:03.1177   7.6746  USER       1.7996 CM*CPU mwds*sec     =     0.5399 RT seconds (  8%)
11:18:03.1180   7.6746  USER       1.8168 CM*I/O requests     =     0.5450 RT seconds (  8%)
11:18:03.1183   7.6746  USER                                     -----------------------
11:18:03.1186   7.6747  USER                         Total    =     6.9807 RT seconds
11:18:03.1189   7.6747  USER
11:19:03.1192   7.6747  USER     RT cost at bid priority 2    = $     0.97
```

ORIGINAL PAGE IS
OF POOR QUALITY

Digital Equipment Corporation - VAX/VMS Version V4.7

NETNONPRIV

OUT1

CPR;1

File DBA0:[]OUT1.CPR;1 (363,61,0), last revised on 22-NOV-1988 09:10, is a 2 block sequential file owned by UIC [11,11]. The records are variable length with FORTRAN (FTN) carriage control. The longest record is 25 bytes.

Job OUT1 (201) queued to SYS$SBPRT on 22-NOV-1988 09:10 by user NETNONPRIV, UIC [11,11], under account 20100ADD at priority 100, started on printer _TTF7: on 22-NOV-1988 09:10 from queue TTF7.

Digital Equipment Corporation - VAX/VMS Version V4.7

(E12.4,1X,E12.4)

0.2724E+01	0.0000E+00
0.2846E+01	0.5049E-01
0.2967E+01	0.1355E+00
0.3089E+01	0.2417E+00
0.3210E+01	0.3572E+00
0.3332E+01	0.4706E+00
0.3453E+01	0.5719E+00
0.3574E+01	0.6542E+00
0.3696E+01	0.7124E+00
0.3817E+01	0.7429E+00
0.3939E+01	0.7450E+00
0.4060E+01	0.7193E+00
0.4182E+01	0.6684E+00
0.4303E+01	0.5957E+00
0.4424E+01	0.5065E+00
0.4546E+01	0.4067E+00
0.4667E+01	0.3027E+00
0.4789E+01	0.2019E+00
0.4910E+01	0.1118E+00
0.5031E+01	0.4101E-01
0.5153E+01	0.0000E+00

FFFFFFFFFF 1111111111111111111111111111 Digital Equipment Corporation - VAX/VMS Version V4.7 1111111111111111111111 FFFFFFFFFF

NETNONPRIV

OUT2

CPR;1

ORIGINAL PAGE IS
OF POOR QUALITY

File _DBA0:[]OUT2.CPR;1 (383,877,0), last revised on 22-NOV-1988 09:10, is a 2 block sequential file owned by UIC [11,11]. The records are variable length with FORTRAN (FTN) carriage control. The longest record is 25 bytes.

Job OUT2 (202) queued to SYS$SBPRT on 22-NOV-1988 09:10 by user NETNONPRIV, UIC [11,11], under account 20120400 at priority 100, started on printer _TTF6: on 22-NOV-1988 09:10 from queue TTF6.

FFFFFFFFFF 1111111111111111111111111111 Digital Equipment Corporation - VAX/VMS Version V4.7 1111111111111111111111 FFFFFFFFFF

```
(E12.4,1X,E12.4)
 0.2724E+01  0.0000E+00
 0.2846E+01  0.3063E-02
 0.2967E+01  0.1433E-01
 0.3089E+01  0.3723E-01
 0.3210E+01  0.7362E-01
 0.3332E+01  0.1239E+00
 0.3453E+01  0.1872E+00
 0.3574E+01  0.2616E+00
 0.3696E+01  0.3446E+00
 0.3817E+01  0.4329E+00
 0.3939E+01  0.5232E+00
 0.4060E+01  0.6121E+00
 0.4182E+01  0.6964E+00
 0.4303E+01  0.7731E+00
 0.4424E+01  0.8400E+00
 0.4546E+01  0.8933E+00
 0.4667E+01  0.9383E+00
 0.4789E+01  0.9672E+00
 0.4910E+01  0.9862E+00
 0.5031E+01  0.9973E+00
 0.5153E+01  0.1000E+01
```

8.0 APPENDIX C

IMSL SUBROUTINE CALLS FROM RANDOM2

1. RNSET - Initializes a random seed for use in the IMSL random number generators.
2. RNNOR - Generates pseudorandom numbers from a standard normal distribution using an inverse CDF method.
3. RNLNL - Generates pseudorandom numbers from a lognormal distribution.
4. DESPL - Performs nonparametric probability density function estimation by the penalized likelihood method.
5. GCDF - Evaluates a general continuous cumulative distribution function given ordinates of the density.

9.0 APPENDIX D

SAMPLE SAS/GRAPH (VER. 5.16) PROGRAM FOR RANDOM2

```
data a;
INFILE 'OUT1.CPR' FIRSTOBS=2;input x y;
GOPTIONS DEVICE=HP7470;
proc gplot;
   axis1 label=(h=1 f=simplex 'LOG OF CYCLES')
         value=(h=1 f=simplex);
   axis2 value=(h=1 f=simplex) label=none;
   plot y*x / haxis=axis1 vaxis=axis2;
   TITLE H=1 A=90 F=SIMPLEX 'PROBABILITY DENSITY FUNCTION';
   symbol i=spline v=square;
data B;
INFILE 'OUT2.CPR' FIRSTOBS=2;input x y;
proc gplot;
   axis1 label=(h=1 f=simplex 'LOG OF CYCLES')
         value=(h=1 f=simplex);
   axis2 value=(h=1 f=simplex) label=none;
   plot y*x / haxis=axis1 vaxis=axis2;
   TITLE H=1 A=90 F=SIMPLEX 'CUMULATIVE DISTRIBUTION FUNCTION';
   symbol i=spline v=square;
```

6.0 APPENDIX 2

FATIGUE STRENGTH DEGRADATION MODEL: RANDOM3 AND RANDOM4 USER MANUAL

FATIGUE STRENGTH REDUCTION MODEL: RANDOM3 and RANDOM4 USER MANUAL

Prepared by :

Lola Boyce, Ph.D., P.E.
Thomas B. Lovelace

APPENDIX 2
of Annual Report
of Project Entitled
Development of Advanced Methodologies
for Probabilistic Constitutive Relationships
of Material Strength Models

NASA Grant No. NAG 3-867

Prepared for :

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Lewis Research Center
Cleveland, OH 44135

The Division of Engineering
The University of Texas at San Antonio
San Antonio, TX 78285
January, 1989

TABLE OF CONTENTS

1.0 INTRODUCTION

This User Manual documents the FORTRAN programs RANDOM3 and RANDOM4. They are based on fatigue strength reduction, using a probabilistic constitutive model. They predict the random lifetime of an engine component to reach a given fatigue strength (see Section 2.0, Theoretical Background).

Included in this Manual are details regarding the theoretical backgrounds of RANDOM3 and RANDOM4, input data instructions and sample problems illustrating the use of RANDOM3 and RANDOM4 . Appendix A gives information on the physical quantities, their symbols, FORTRAN names and both SI and U.S. Customary units. Appendix B and C include photocopies of the actual computer printout corresponding to the sample problems. Appendices D and E detail the IMSL, Version 10 [1], subroutines and functions called by RANDOM3 and RANDOM4 and SAS/GRAPH [2] programs that can be used to plot both the probability density functions (p.d.f.) and the cumulative distribution functions (c.d.f.).

2.0 THEORETICAL BACKGROUND

Fatigue strength data are usually presented as cycles to failure for each of several stress amplitudes, the familiar S-N diagram. Results indicate that for lower stress amplitudes the cycles (or time) to failure increases. Thus, a power curve fit through the data yields a monotonically decreasing curve. In general, this curve is represented as

$$S = [N/C']^{-1/m'} \tag{6}$$

where the primitive variables in this equation are as follows: S is the applied constant amplitude alternating stress at failure or fatigue strength, N is number of cycles, C' is a material parameter that varies from specimen to specimen and m' is a material constant.[3] Equation (6) can be written in terms of "cycles to reach a given fatigue strength" as

$$N = C'S^{-m'} \tag{7}$$

Recently another fatigue strength reduction model has been proposed that takes into account the effect of temperature as well as other parameters that affect strength.[4] The general form of the constitutive relationships for this model is applied to the constituents of high temperature composite materials. Specifically, it is applied herein for the case of a single material constituent. The mechanical property of interest is fatigue strength which is expressed in terms of primitive variables, including the general categories of temperature, mechanical cycles and mean stress. For these categories, the relationship becomes

$$\frac{S}{S_o} = \left[\frac{T_F - T}{T_F - T_o}\right]^n \left[\frac{S_F - \sigma}{S_F - \sigma_o}\right]^m \left[\frac{\log N_{MF} - \log N_M}{\log N_{MF} - \log N_{MO}}\right]^q \tag{8}$$

where S is the applied constant amplitude alternating stress at failure (fatigue strength) at current (or operating) temperature, T, mean stress, σ, and mechanical cycle, N_M. S_O is fatigue strength at reference temperature, T_O (usually room temperature), reference mean stress (or residual stress), σ_O, and reference mechanical cycle, N_{MO}. Also, T_F is the final or melting temperature of the material, S_F is the final or tensile strength of the material, and N_{MF} is the final mechanical cycle or lifetime. Empirical parameters, n, m, and q, are determined from available experimental data or estimated from anticipated behavior of the particular product term.[5] Note that the term containing mechanical cycles is expressed in terms of the log of cycles rather than cycles. This formulation is attractive when N_M and N_{MO} are small compared to N_{MF}. The equation may be solved for N_M, or the "cycles to reach a given fatigue strength." The expression is

$$N = 10 \exp\left[\log N_{MF} - \left[(\log N_{MF} - \log N_{MO})\left[\frac{S}{S_o\left[\frac{T_F - T}{T_F - T_o}\right]^n \left[\frac{S_F - \sigma}{S_F - \sigma_o}\right]^m}\right]^{1/q}\right]\right] \tag{9}$$

For values typical of a cast nickel base-superalloy subjected to typical loads and temperatures, equation (9) indicates increasing life for decreasing temperature, decreasing tensile mean stress, and decreasing applied alternating stress. It indicates decreasing life for increasing temperature, decreasing compressive mean stress, and increasing applied alternating stress. Therefore, equation (9) predicts observed trends in general.

Probabilistic analysis, via simulation, yields the distribution of the dependent random variable, cycles, N. A probability density function (p.d.f.) of cycles is generated using the maximum penalized likelihood method for RANDOM3. For RANDOM4, a p.d.f. of cycles is generated using the maximum entropy method. Maximum entropy uses Jaynes' principle which says that "the minimally prejudiced distribution is that which maximizes the entropy subjected to the constraints supplied by the given information."[6]

3.0 INPUT DATA

Data input for RANDOM3 and RANDOM4 is user friendly and easy to manipulate (see, for example, the file entitled NORMAL.INP, in Section 4.0). The first twelve lines of input have the same format, 2E12.4 and the last two lines differ. The last two lines of input have the formats I3,2X,I3,2X,2E12.4,2X,I3 and I3, respectively. A brief, line by line description is given along with an example for each line (NOTE: the ruler is to aid the user in formatting and is not a part of the input). A table listing the physical quantities, their units and symbols is given in Appendix A.

1. Random Number Generator Seed, ISEED, and Sample Size, NTOT

EXAMPLE:

```
123456789012345678901234567890
                  1                  40
```

2. Ultimate Tensile Strength, SF

EXAMPLE:

```
123456789012345678901234567890
      900.0000       45.0000
```

3. Log of Final Cycle, NMF

EXAMPLE:

```
123456789012345678901234567890
        8.0000        0.8000
```

4. Reference Fatigue Strength, SO

EXAMPLE:

```
123456789012345678901234567890
      500.0000       25.0000
```

5. Log of Reference Cycle, NMO

EXAMPLE:

```
123456789012345678901234567890
        7.0000        0.7000
```

6. Current Fatigue Strength, S

EXAMPLE:

```
123456789012345678901234567890
        250.0000       12.0000
```

7. Residual Compressive Stress, SIGO

EXAMPLE:

```
123456789012345678901234567890
         20.0000        1.0000
```

8. Current Mean Stress, SIG

EXAMPLE:

```
123456789012345678901234567890
        150.0000        7.5000
```

9. Temperature Exponent, XXN, Stress Exponent, XXM, and Cycle Exponent, XXQ

EXAMPLE:

```
123456789012345678901234567890
          0.5000        0.0150
```

10. Melting Temperature, TF

EXAMPLE:

```
123456789012345678901234567890
       1500.0000       75.0000
```

11. Reference Temperature, TO

EXAMPLE:

```
123456789012345678901234567890
         20.0000        0.6000
```

12. Current Temperature, T

EXAMPLE:

```
1234567890123456789012345678901234567890
         850.0000      25.0000
```

13. The DESPL[1] parameters are NODE, INIT, ALPHA, EPS, and MAXIT and are entered in that order as follows:

EXAMPLE:

```
12345678901234567890123456789012345678901234567890
  21     0          20.0000       1.0E-05    30
```

14. The DESPL parameter, IOPT, is entered as follows:

EXAMPLE:

```
1234567890
   2
```

4.0 SAMPLE PROBLEMS FOR RANDOM3 AND RANDOM4

The objective of these programs is to predict the random lifetime to reach a given fatigue strength for an engine component. The theory is based on fatigue strength reduction, using a probabilistic constitutive model. The only difference between RANDOM3 and RANDOM4 is the method used to generate p.d.f. estimates. RANDOM3 uses maximum penalized likelihood, while RANDOM4 uses maximum entropy (see Section 2.0, Theoretical Background). RANDOM3 and RANDOM4 input parameters are given in Table A2.1.

TABLE A2.1 RANDOM3 and RANDOM4 input (SI units)

FORTRAN Name	Distribution Type	Mean	Standard Deviation (Value)	Standard Deviation (% of Mean)
SF	normal	900.0	45.0	(3%)
NMF	lognormal	8.0	0.8	(10%)
SO	lognormal	500.0	25.0	(5%)
NMO	lognormal	7.0	0.7	(10%)
S	lognormal	250.0	12.5	(5%)
SIGO	lognormal	-20.0	-1.0	(1%)
SIG	lognormal	150.0	7.5	(5%)
XXN	normal	0.5	0.015	(0.3%)
XXM	normal	0.5	0.015	(0.3%)
XXQ	normal	0.5	0.015	(0.3%)
TF	normal	1500.0	45.0	(3%)
TO	normal	20.0	0.6	(3%)
T	normal	850.0	25.5	(3%)

The input is entered in the following format in a file entitled NORMAL.INP.

```
1234567890123456789012345678901234567890
----------------------------------------
           1          40
    900.0000     45.0000
      8.0000      0.8000
    500.0000     25.0000
      7.0000      0.7000
    250.0000     12.5000
     20.0000      1.0000
    150.0000      7.5000
      0.5000      0.0150
   1500.0000     75.0000
     20.0000      0.6000
    850.0000     25.5000
 21    0         20.00       1.0E-05    30
  2
```

Execution of RANDOM3 and RANDOM4 (source code entitled NR3.FOR and NR4.FOR, respectively) produces files entitled RANDM33 and RANDM44. These give intermediate results (see Appendices B and C). Execution also produces plotfiles entitled PLOT1 and PLOT2 (see Appendices B and C). These files are used to plot the X and Y axes of the probability density function (p.d.f.) and the cumulative distribution function (c.d.f.), respectively, generated by RANDOM3 and RANDOM4. The plots are drawn from the plotfiles by the SAS/GRAPH graphing program (see Appendix D). These plots for the sample problem are shown Figures 1, 2, 3, and 4. This same sample problem has been reported in Boyce and Chamis.[7] There, however, it utilized U.S. Customary units and older versions of RANDOM3 and RANDOM4 (using IMSL Version 9.2 subroutines).

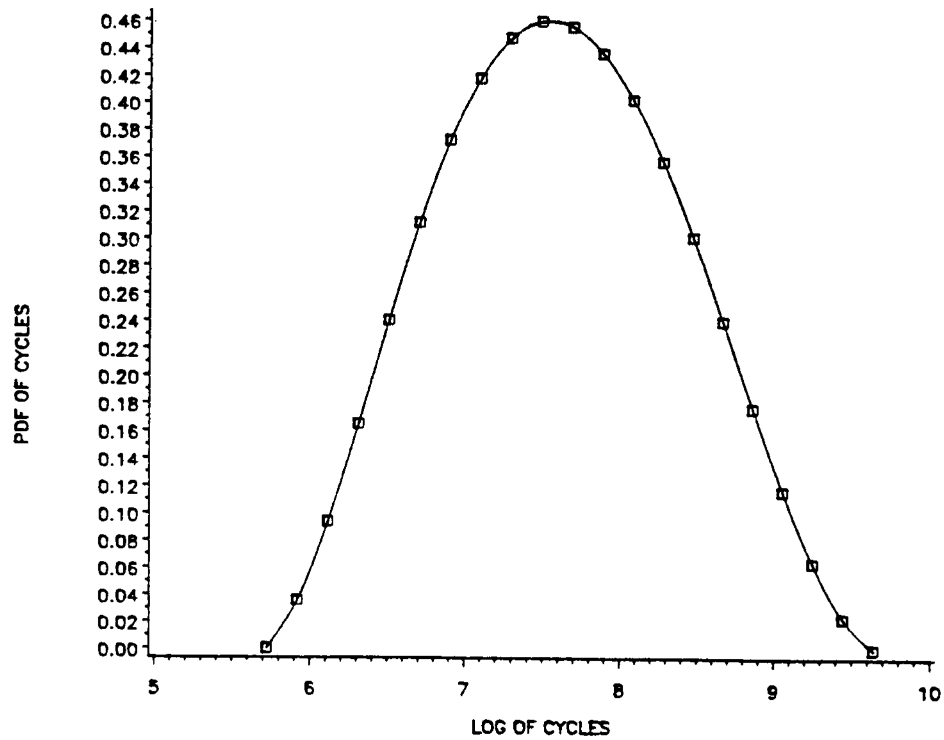

Fig. A2.1 p.d.f. of log of mechanical cycles for fatigue strength reduction model, using maximum penalized likelihood method of p.d.f. generation.

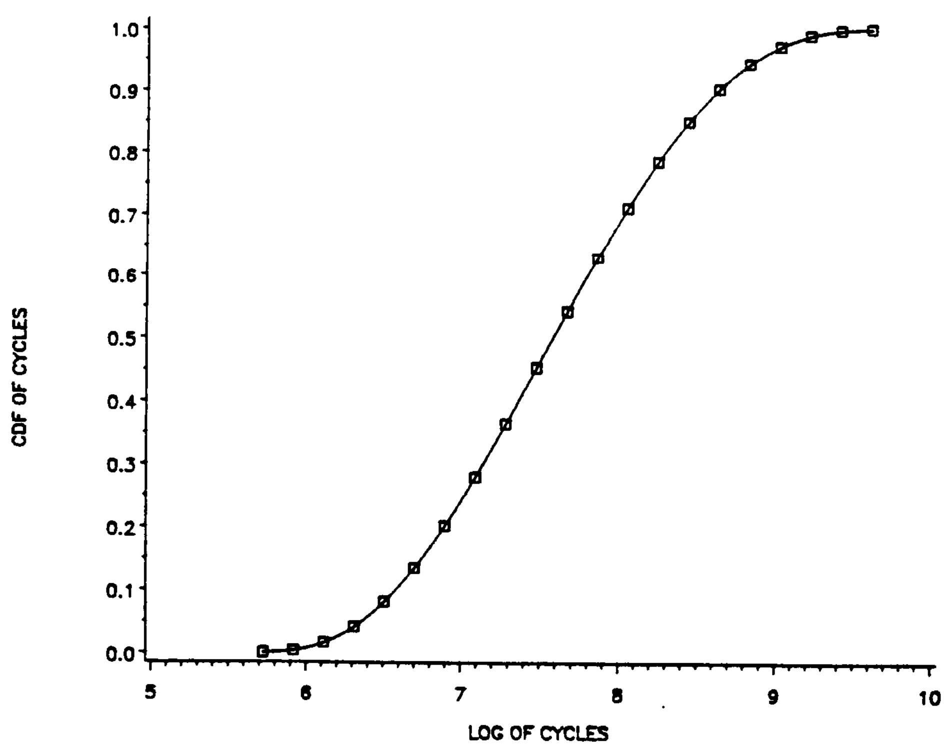

Fig. A2.2 c.d.f. of log of mechanical cycles for fatigue strength reduction model, using maximum penalized likelihood method of p.d.f. generation.

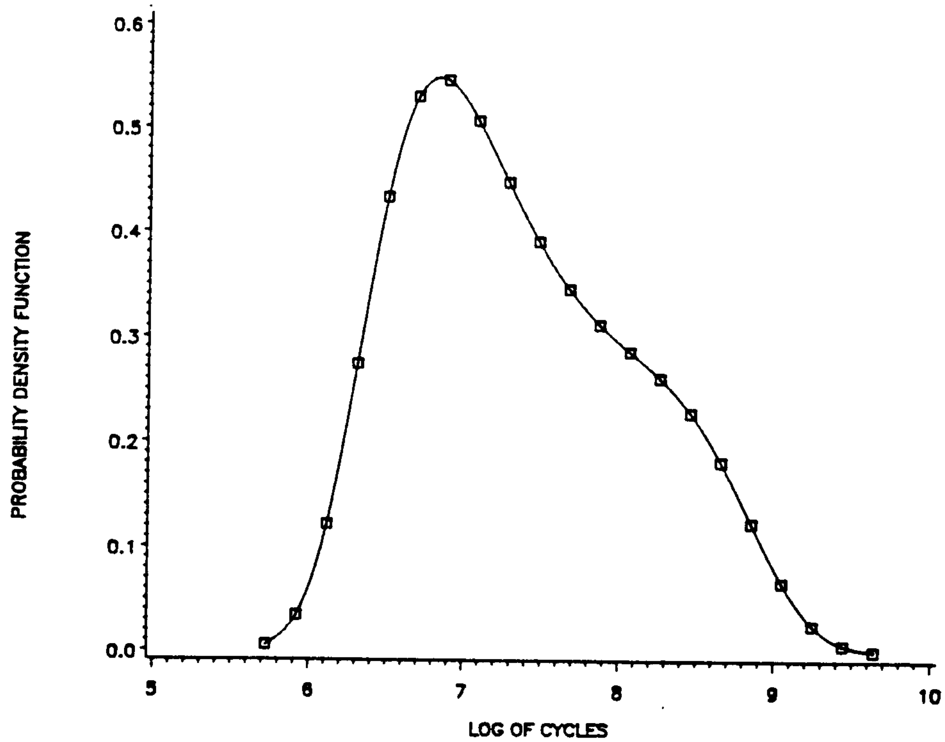

Fig. A2.3 p.d.f. of log of mechanical cycles for fatigue strength reduction model, using maximum entropy method of p.d.f. generation.

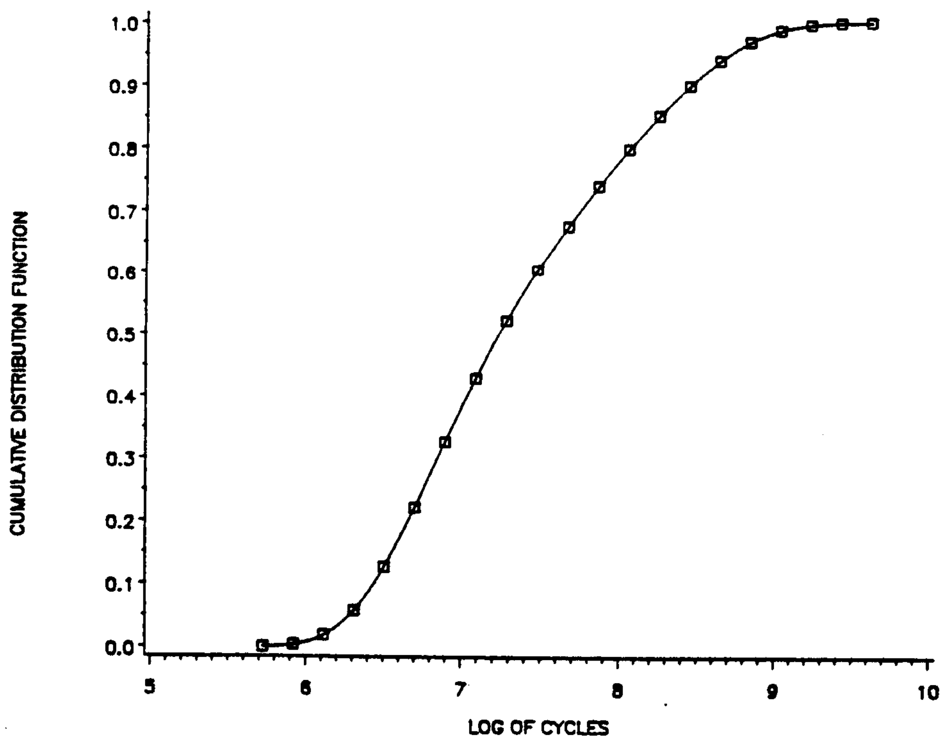

Fig. A2.4 c.d.f. of log of mechanical cycles for fatigue strength reduction model, using maximum entropy method of p.d.f. generation.

5.0 REFERENCES

[1] IMSL, "STAT/LIBRARY, FORTRAN Subroutines for Statistical Analysis", Houston, Texas

[2] SAS Institute, Inc., SAS/GRAPH User's Guide, Version 5 Edition, Cary NC: SAS Institute, Inc., 1985, 596 pp.

[3] Madsen, H.O., "Bayesian Fatigue Life Prediction," Probabilistic Methods in the Mechanics of Solids and Structures, S. Eddwertz and N.C. Lind, Eds., Proceedings of the IUTAM Symposium, Stockholm, Sweden, 1984,pp. 395-406.

[4] Hopkins, D.A. and Chamis, C.C., "A Unique Set of Micromechanics Equations for High Temperature Metal Matrix Composites," NASA TM87154, Nov., 1985.

[5] Chamis, C.C. and Hopkins, D.A., "Thermoviscoplastic Nonlinear Constitutive Relationships for Structural Analysis of High Temperature Metal Matrix Composites," NASA TM 87291, Nov., 1985.

[6] Siddall, J.N., "A Comparison of Several Methods of Probabilistic Modeling," Proceedings of the Computers in Engineering Conference, ASME, San Diego, CA, Vol. 4, 1982, pp. 231-238.

[7] Boyce, L. and Chamis, C.C., "Probabilistic Constitutive Relations for Cyclic Material Strength Models," Proceedings, 29th Structures, Structural Dynamics and Materials Conference, Williamsburg, VA, 1988.

6.0 APPENDIX A

PHYSICAL QUANTITIES, SYMBOLS, AND UNITS

The physical quantities, their symbols and units for the fatigue crack growth model are given in the following table.

Table A2.2 Physical quantities, symbols, and units for fatigue crack growth model for RANDOM3 and RANDOM4.

Physical Quantity	Theory Symbol	FORTRAN Name	Units SI	Units U.S.
Ultimate Tensile Strength	SF	SF	MPa	ksi
Final Cycle (lifetime)	N_{MF}	NMF	dimensionless	
Reference Fatigue Strength	SO	SO	MPa	ksi
Reference Cycles	N_{MO}	NMO	dimensionless	
Current Fatigue Strengths	S	S	MPa	ksi
Residual Compressive Stress	σ_o	SIGO	MPa	ksi
Current Mean Stress	σ	SIG	MPa	ksi
	n	XXN	dimensionless	
Empirical Material Parameters	m	XXM	dimensionless	
	q	XXQ	dimensionless	
Melting Temperature	TF	TF	°C	°F
Reference Temperature	TO	TO	°C	°F
Current Temperature	T	T	°C	°F

7.0 APPENDIX B

RANDOM3 SAMPLE PROBLEM: SOURCE, INPUT AND OUTPUT FILES

CCCCCCCCCC 22222222222222222222222222 Digital Equipment Corporation - VAX/VMS Version V4.7 22222222222222222222222222 CCCCCCCC

DECNET

NR3

FOR;3

File _DUA0:[]NR3.FOR;3 (252,112,0), last revised on 22-DEC-1988 13:08, is a 48 block sequential file owned by UIC [DECNET]. The records are variable length with implied (CR) carriage control. The longest record is 72 bytes.

Job NR3 (130) queued to TERM$LA120A on 22-DEC-1988 13:09 by user DECNET, UIC [DECNET], under account DECNET at priority 100, started on printer LTA4: on 22-DEC-1988 13:09 from queue TERM$LA120A.

CCCCCCCCCC 22222222222222222222222222 Digital Equipment Corporation - VAX/VMS Version V4.7 22222222222222222222222222 CCCCCCCC

```
JOB,JN=COMP3,US=USAQ530,RT=60,T=30,MFL=3000000.
ACCOUNT,UPW=LOLAB.
DELETE,PDN=NR3BLD,ID=SMBOYCE.
CFT77,LIST.
REWIND,DN=$BLD.
BUILD,I=0,OBL=0.
SAVE,DN=$NBL,PDN=NR3BLD,ID=SMBOYCE.
DELETE,PDN=NR3BLD,ID=SMBOYCE,ED=-1.
/EOF
C CHAMIS MICROMECHANICS CONSTITUTIVE EQUATIONS;
C RANDOMIZED AND APPLIED TO FATIGUE STRENGTH
      INTEGER NTOT, ISEED, M, INIT, NMISS, MAXIT, NODE
      REAL XM, XS, YM, YS, EPS, P, RWKSP(7612), ALPHA
      COMMON /WORKSP/ RWKSP
      DIMENSION SF(10000),XLNMF(10000),SO(10000)
      DIMENSION XLNMO(10000),S(10000)
      DIMENSION SIGO(10000),SIG(10000)
      DIMENSION XXN(10000),XXM(10000),XXQ(10000)
      DIMENSION XNM(10000),BNDSX(10000)
      DIMENSION TF(10000),TO(10000),T(10000)
      DIMENSION STAT(10000), DENS(10000),DISTX(10000)
      DIMENSION BNDS(10000),BB(999),FF(999)
      DIMENSION XX(999),PP(999)
      DIMENSION XXXX(999),PPPP(999)
      DIMENSION BBBB(999),FFFF(999)
      DIMENSION C(10000)
      EXTERNAL RNLNL, RNSET, RNNOR, DESPL, IWKIN
 1001 FORMAT(5E12.4)
 1002 FORMAT(I12,I12)
 1003 FORMAT(I4,I4)
 1004 FORMAT(I4)
 1009 FORMAT(I3,2X,I3,2X,2E12.4,2X,I3)
 1010 FORMAT(I3)
 1011 FORMAT(2E12.4)
C LOGNORMAL ULTIMATE TENSILE STRENGTH, SF
      READ (3,1002) ISEED,NTOT
      WRITE(6,1002) ISEED,NTOT
      READ (3,1011) XM,XS
      WRITE(6,1011) XM,XS
      YS = SQRT(LOG(1.0+(XS/XM)**2.))
      YM = LOG(XM) - 0.5*YS**2.
      CALL RNSET(ISEED)
      CALL RNLNL(NTOT,YM,YS,SF)
      WRITE(6,2020)
 2020 FORMAT(' LOGNORMAL SF')
      WRITE(6,1001)(SF(I),I=1,NTOT)
C LOGNORMAL LOG OF FINAL CYCLE, XLNMF
      WRITE(6,1002) ISEED,NTOT
      READ (3,1011) XM,XS
      WRITE(6,1011) XM,XS
      YS = SQRT( LOG(1.0+(XS/XM)**2))
      YM = LOG(XM) - 0.5*YS**2
      CALL RNSET(ISEED)
      CALL RNLNL(NTOT,YM,YS,XLNMF)
      WRITE(6,2021)
 2021 FORMAT(' LOGNORMAL XLNMF')
      WRITE( 6,1001) (XLNMF(I),I=1,NTOT)
C LOGNORMAL FATIGUE STRENGTH AT REFERENCE CONDITIONS, SO
      WRITE(6,1002) ISEED,NTOT
      READ (3,1011) XM, XS
      WRITE (6,1011) XM, XS
      YS = SQRT( LOG(1.0+(XS/XM)**2))
      YM = LOG(XM) - 0.5*YS**2
      CALL RNSET(ISEED)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      CALL RNLNL(NTOT,YM,YS,SO)
      WRITE(6,2022)
 2022 FORMAT(' LOGNORMAL SO')
      WRITE( 6,1001) (SO(I),I=1,NTOT)
C LOGNORMAL LOG OF REFERENCE CYCLES, XLNMO
      WRITE(6,1002) ISEED,NTOT
      READ (3,1011) XM, XS
      WRITE(6,1011) XM, XS
      YS = SQRT(LOG(1.0+(XS/XM)**2))
      YM = LOG(XM) - 0.5*YS**2
      CALL RNSET(ISEED)
      CALL RNLNL(NTOT,YM,YS,XLNMO)
      WRITE (6,2023)
 2023 FORMAT (' LOGNORMAL XLNMO')
      WRITE (6,1001)(XLNMO(I),I=1,NTOT)
C LOGNORMAL FATIGUE STRENGTH AT CURRENT CONDITIONS, S
      WRITE(6,1002) ISEED,NTOT
      READ (3,1011) XM, XS
      WRITE(6,1011) XM, XS
      YS = SQRT(LOG(1.0+(XS/XM)**2.))
      YM = LOG(XM) - 0.5*YS**2.
      CALL RNSET(ISEED)
      CALL RNLNL(NTOT,YM,YS,S)
      WRITE (6,2024)
 2024 FORMAT(' LOGNORMAL S')
      WRITE(6,1001)(S(I),I=1,NTOT)
C DEFINE RANDOM STRESSES
C LOGNORMAL REFERENCE STRESS, SIGO
      WRITE(6,1002)ISEED,NTOT
      READ (3,1011) XM, XS
      WRITE(6,1011) XM, XS
      YS=SQRT(LOG(1.0+(XS/XM)**2.))
      YM=LOG(XM) - 0.5*YS**2.
      CALL RNSET(ISEED)
      CALL RNLNL(NTOT,YM,YS,SIGO)
C CHANGE SIGO TO NEGATIVE VALUES FOR COMPRESSIVE
C RESIDUAL STRESSES
      DO 201 I = 1,NTOT
      SIGO(I)=-SIGO(I)
  201 CONTINUE
      WRITE(6,2036)
 2036 FORMAT(' LOGNORMAL SIGO')
      WRITE(6,1001)(SIGO(I),I=1,NTOT)
C LOGNORMAL CURRENT STRESS, SIG
      WRITE(6,1002)ISEED,NTOT
      READ (3,1011) XM, XS
      WRITE(6,1011) XM, XS
      YS=SQRT(LOG(1.0+(XS/XM)**2.))
      YM=LOG(XM) - 0.5*YS**2.
      CALL RNSET(ISEED)
      CALL RNLNL(NTOT,YM,YS,SIG)
      WRITE(6,2037)
 2037 FORMAT(' LOGNORMAL SIG')
      WRITE(6,1001)(SIG(I),I=1,NTOT)
C NORMAL EXPONENTS, XXN,XXM,XXQ
      WRITE(6,1002)ISEED,NTOT
      READ (3,1011) YM, YS
      WRITE(6,1011) YM, YS
      CALL RNSET(ISEED)
      CALL RNNOR(NTOT,XXN)
      DO 202 I=1,NTOT
      XXN(I)=YS*XXN(I)+YM
  202 CONTINUE
      WRITE(6,2025)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
 2025 FORMAT(' NORMAL XXN')
      WRITE(6,1001)(XXN(I),I=1,NTOT)
      WRITE(6,1002)ISEED,NTOT
      CALL RNSET(ISEED)
      CALL RNNOR(NTOT,XXM)
      DO 203 I=1,NTOT
      XXM(I)=YS*XXM(I)+YM
  203 CONTINUE
      WRITE(6,2026)
 2026 FORMAT(' NORMAL XXM')
      WRITE(6,1001)(XXM(I),I=1,NTOT)
      WRITE(6,1002)ISEED,NTOT
      CALL RNSET(ISEED)
      CALL RNNOR(NTOT,XXQ)
      DO 204 I=1,NTOT
      XXQ(I)=YS*XXQ(I)+YM
  204 CONTINUE
      WRITE(6,2027)
 2027 FORMAT(' NORMAL XXQ')
      WRITE(6,1001)(XXQ(I),I=1,NTOT)
C NORMAL TEMPERATURES, TF,TO,T
C NORMAL FINAL (MELTING) TEMPERATURE, TF
      WRITE(6,1002)ISEED,NTOT
      READ(3,1011) YM, YS
      WRITE(6,1011) YM, YS
      CALL RNSET(ISEED)
      CALL RNNOR(NTOT,TF)
      DO 205 I=1,NTOT
      TF(I)=YS*TF(I)+YM
  205 CONTINUE
      WRITE(6,2046)
 2046 FORMAT(' NORMAL TF')
      WRITE(6,1001)(TF(I),I=1,NTOT)
C NORMAL REFERENCE TEMPERATURE, TO
      WRITE(6,1002)ISEED,NTOT
      READ (3,1011) YM, YS
      WRITE(6,1011) YM, YS
      CALL RNSET(ISEED)
      CALL RNNOR(NTOT,TO)
      DO 206 I=1,NTOT
      TO(I)=YS*TO(I)+YM
  206 CONTINUE
      WRITE(6,2047)
 2047 FORMAT(' NORMAL TO')
      WRITE(6,1001)(TO(I),I=1,NTOT)
C NORMAL CURRENT TEMPERATURE,T
      WRITE(6,1002)ISEED,NTOT
      READ(3,1011) YM, YS
      WRITE(6,1011) YM, YS
      CALL RNSET(ISEED)
      CALL RNNOR(NTOT,T)
      DO 207 I=1,NTOT
      T(I)=YS*T(I)+YM
  207 CONTINUE
      WRITE(6,2048)
 2048 FORMAT(' NORMAL T')
      WRITE(6,1001)(T(I),I=1,NTOT)
C CALCULATE LOG OF CURRENT CYCLES, LOG XNM
      DO 102 I=1,NTOT
      RS=((SF(I)-SIG(I))/(SF(I)-SIGO(I)))**XXM(I)
      TEMP=((TF(I)-T(I))/(TF(I)-TO(I)))**XXN(I)
      XNM1=(S(I)/(SO(I)*TEMP*RS))**(1./XXQ(I))
      XNM2=(XLNMF(I)-((XLNMF(I)-XLNMO(I))*XNM1))
      IF(XNM2.LT.0.0)XNM2=0.0
```

OF POOR QUALITY

```
      XNM(I)=XNM2
  102 CONTINUE
      WRITE(6,2028)
 2028 FORMAT(' LOG OF CYCLES TO REACH MEAN FATIGUE STR =    ',/,
     1'250 MPA')
      WRITE(6,1001)(XNM(I),I=1,NTOT)
C SORT LOG OF CYCLES
      CALL SORT(XNM,NTOT)
      WRITE(6,2029)
 2029 FORMAT(' SORTED LOG OF CYCLES')
      WRITE(6,1001)(XNM(I),I=1,NTOT)
C CALCULATE PDF OF LOG OF CURRENT CYCLES,LOG XNM
      READ(3,1009)NODE,INIT,ALPHA,EPS,MAXIT
      WRITE(6,985)
  985 FORMAT(' DESPL PARAMETERS')
      WRITE(6,1009)NODE,INIT,ALPHA,EPS,MAXIT
      BNDS(1)=XNM(1) - 0.05*XNM(1)
      BNDS(2)=XNM(NTOT) + 0.05*XNM(NTOT)
      WRITE(6,979)BNDS(1),BNDS(2)
  979 FORMAT(' BNDS(1),BNDS(2)=',E12.4,1X,E12.4)
      CALL DESPL(NTOT,XNM,NODE,BNDS,INIT,ALPHA,MAXIT,EPS,DENS,STAT,
     1NMISS)
      WRITE(6,980)
  980 FORMAT(' PDF OF LOG OF CURRENT CYCLES,LOG XNM,Y AXIS OF PDF PLOT')
      WRITE(6,1001)(DENS(I),I=1,NODE)
      WRITE(6,981)
  981 FORMAT(' OUTPUT STATISTICS')
      WRITE(6,1001)(STAT(I),I=1,4)
      WRITE(6,982)
  982 FORMAT(' NUMBER OF MISSING VALUES')
      WRITE(6,1010)NMISS
C
C CALCULATE WINDOW WIDTH, HH
C
      HH=(BNDS(2)-BNDS(1))/(NODE-1)
C
C CALCULATE VALUES OF LOG OF CURRENT CYCLES AT WHICH PDF IS ESTIMATED;
C ALSO CALLED 'NODE' VALUES
C
      DO 6001,I=1,NODE-2
      BNDS(I+2)=BNDS(1) + (I*HH)
 6001 CONTINUE
      WRITE(6,983)
  983 FORMAT(' LOG OF CURRENT CYCLES, LOG XNM')
      WRITE(6,1001)(BNDS(I),I=1,NODE)
C
C REORDER BNDS FOR PLOTTING
C
      SAVE1 = BNDS(2)
      SAVE2 = BNDS(NODE)
      BNDS(NODE)=BNDS(2)
      DO 6002, I=1, NODE-2
      BNDS(I+1)=BNDS(I+2)
 6002 CONTINUE
      BNDS(NODE-1)=SAVE2
      BNDS(NODE)=SAVE1
      WRITE(6,984)
  984 FORMAT(' ORDERED LOG OF CURRENT CYCLES, LOG,XNM,
     1X AXIS PDF, CDF PLOT')
      WRITE(6,1001)(BNDS(I),I=1,NODE)
C WRITE LOG OF CURRENT CYCLES AND PDF OF LOG OF CURRENT CYCLES,
C   LOG XNM TO PLOT FILES
      WRITE(34,990)
  990 FORMAT(' (E12.4,1X,E12.4)')
```

```
      WRITE(34,991)(BNDS(J),DENS(J),J=1,NODE)
  991 FORMAT(E12.4,1X,E12.4)
C
C CALCULATE CDF OF LOG OF CURRENT CYCLES
C
      READ(3,1010)IOPT
      WRITE(6,992)
  992 FORMAT(' GCDF PARAMETERS')
      WRITE(6,1010)IOPT
      XO=BNDS(1)
      DO 6003,I=1,NODE
      P=GCDF(XO,IOPT,NODE,BNDS,DENS)
      BNDSX(I)=XO
      XO=XO+HH
      DISTX(I)=P
 6003 CONTINUE
      WRITE(6,994)
  994 FORMAT(' CDF OF LOG OF CURRENT CYCLES, LOG XNM,
     1Y AXIS OF PDF, CDF PLOT')
      WRITE(6,1001)(DISTX(I),I=1,NODE)
C
      WRITE(6,993)
  993 FORMAT(' ORDERED LOG OF CURRENT CYCLES, LOG XNM,
     1X AXIS OF PDF, CDF PLOT')
      WRITE(6,1001)(BNDS(I),I=1,NODE)
      WRITE(6,1001)(BNDSX(I),I=1,NODE)
C
C WRITE LOG OF CURRENT CYCLES AND CDF OF LOG OF CURRENT
C TO THE PLOT FILES
C
      WRITE(35,990)
      WRITE(35,991)(BNDS(J),DISTX(J),J=1,NODE)
      STOP
      END
      SUBROUTINE SORT(Y,N)
      DIMENSION Y(10000)
      N1=N-1
      DO 1 I=1, N1
      J=I+1
      DO 2 K=J, N
      IF (Y(I).LT.Y(K))GO TO 2
      TEMP=Y(I)
      Y(I)=Y(K)
      Y(K)=TEMP
    2 CONTINUE
    1 CONTINUE
      RETURN
      END
C-----------------------------------------------------------------------
C   IMSL Name:  D3SPL/DD3SPL (Single/Double precision version)
C
C   Computer:   IBM/SINGLE
C
C   Revised:    November 1, 1985
C
C   Purpose:    Nonparametric probability density function estimation
C               estimation by the penalized likelihood method.
C
C   Usage:      CALL D3SPL (NOBS, X, NODE, BNDS, INIT, ALPHA, MAXIT, EPS,
C                           DENS, STAT, HESS, LDHESS, ILOHI, DENEST, B,
C                           IPVT, WK2)
C
C   Arguments:
C      NOBS   - Number of observations.  (Input)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C     X      - Vector of length NOBS containing the random sample of
C              responses.  (Input)
C     NODE   - Number of mesh nodes for the discrete pdf estimate.
C              (Input)
C     BNDS   - Vector of length 2 containing the minimum and maximum
C              values for X(i) in BNDS(1) and BNDS(2), respectively.
C              (Input)
C     INIT   - Initialization option.  (Input)
C     ALPHA  - Positive penalty weighting factor which controls the
C              smoothness of the estimate.  (Input)
C     MAXIT  - Maximum number of iterations allowed in the iterative
C              procedure.  (Input)
C     EPS    - Convergence criterion.  (Input)
C     DENS   - Vector of length NODE containing the estimated values of
C              the discrete pdf at the NODE equally spaced mesh nodes.
C              (Input/output if INIT=1, Output otherwise)
C     STAT   - Vector of length 4 containing out statistics.  (Output)
C              STAT(1) and STAT(2) contain the log-likelihood and the
C              log-penalty terms, respectively.  STAT(3) and STAT(4)
C              contain the estimated mean and variance for the
C              estimated density.
C     HESS   - Seven by NODE-2 hessian matrix (and its factorization).
C              (Output)
C     LDHESS - Leading dimension of HESS exactly as specified in the
C              dimension statement in the calling program.  (Input)
C     ILOHI  - NODE by 2 matrix containing the indices for the risk set
C              at each node value.  (Output)
C     DENEST - NODE by 3matrix containing the gradient vector, among
C              other quantities.  (Output)
C     B      - Vector of length NODE containing the NODE values.
C              (Output)
C     IPVT   - Pivot vector of length NODE-2.  (Output)
C     WK2    - Work vector of length NODE-2.  (Output)
C
C  Chapter:    STAT/LIBRARY Density and Hazard Estimation
C
C  Copyright:  1985 by IMSL, Inc.  All Rights Reserved.
C
C  Warranty:   IMSL warrants only that IMSL testing has been applied
C              to this code.  No other warranty, expressed or implied,
C              is applicable.
C
C-----------------------------------------------------------------------
C
      SUBROUTINE D3SPL (NOBS, X, NODE, BNDS, INIT, ALPHA, MAXIT, EPS,
     &                  DENS, STAT, HESS, LDHESS, ILOHI, DENEST, B,
     &                  IPVT, WK2)
C                                  SPECIFICATIONS FOR ARGUMENTS
      INTEGER    NOBS, NODE, INIT, MAXIT, LDHESS, ILOHI(NODE,*),
     &           IPVT(*)
      REAL       ALPHA, EPS, X(*), BNDS(2), DENS(*), STAT(4),
     &           HESS(LDHESS,*), DENEST(NODE,*), B(*), WK2(*)
C                                  SPECIFICATIONS FOR LOCAL VARIABLES
      INTEGER    I, IMPTR, IPTR, ITER, K, KM1, KM2, KP1, KP2, M, MOLD,
     &           NER, NOB1
      REAL       BK, BKM1, BSMALL, CK, CKM1, CKM2, CKMCM1, CKP1, CKP2,
     &           CONS, EPS1, FACTOR, FK, FKM1, FKM2, FKP1, H, H2, H3,
     &           SUM, TEMP, WK(4)
      DOUBLE PRECISION SUM1, SUM2, SUM3
C                                  SPECIFICATIONS FOR SAVE VARIABLES
      INTEGER    MINCR(8)
      SAVE       MINCR
C                                  SPECIFICATIONS FOR INIRINSICS
C     intrinsic  alog,amax1,max0,min0,mod,sqrt
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      INTRINSIC  ALOG, AMAX1, MAXO, MINO, MOD, SQRT
      INTEGER    MAXO, MINO, MOD
      REAL       ALOG, AMAX1, SQRT
C                              SPECIFICATIONS FOR SUBROUTINES
      EXTERNAL   E1MES, E1POP, E1PSH, E1STI, E1STR, SADD, SAXPY,
     &           SCOPY, SHPROD, SSCAL, D2SPT, L2TRB, LFSRB
C                              SPECIFICATIONS FOR FUNCTIONS
      EXTERNAL   ISMIN, N1RCD, SDOT, SNRM2, SSUM
      INTEGER    ISMIN, N1RCD
      REAL       SDOT, SNRM2, SSUM
C
      DATA MINCR/5, 9, 17, 33, 65, 129, 253, 100001/
C
      CALL E1PSH ('D3SPL ')
C                              Error checks
      NER = 1
      IF (NOBS .LT. 1) THEN
         CALL E1MES (5, 1, 'After removing all missing (NaN, not a '//
     &               'number) values from X there are no valid '//
     &               'observations.  At least one valid observation '//
     &               'is necessary.')
      END IF
      IF (NODE .LE. 4) THEN
         CALL E1STI (1, NODE)
         CALL E1MES (5, 2, 'NODE = %(I1).  The number of mesh '//
     &               'nodes, NODE, must be an odd integer greater '//
     &               'than 4.')
      ELSE IF (MOD(NODE,2) .EQ. 0) THEN
         CALL E1STI (1, NODE)
         CALL E1MES (5, 3, 'NODE = %(I1) must be an odd integer '//
     &               'greater than 4.')
      END IF
      IF (ALPHA .LE. 0.0) THEN
         CALL E1STR (1, ALPHA)
         CALL E1MES (5, 4, 'ALPHA = %(R1).  The penalty weighting '//
     &               'factor which controls smoothness, ALPHA, must '//
     &               'be greater than 0.')
      END IF
      IF (MAXIT .LE. 0.0) THEN
         CALL E1STI (1, MAXIT)
         CALL E1MES (5, 5, 'MAXIT = %(I1).  The maximum number '//
     &               'of iterations, MAXIT, must be greater than 0.')
      END IF
      IF (BNDS(1) .GT. BNDS(2)) THEN
         CALL E1STR (1, BNDS(1))
         CALL E1STR (2, BNDS(2))
         CALL E1MES (5, 6, 'BNDS(1) = %(R1) and BNDS(2) = '//
     &               '%(R2).  The minimum value for X, BNDS(1), must '//
     &               'be less than or equal to the maximum value for '//
     &               'X, BNDS(2).')
      END IF
      IF (INIT .NE. 0) THEN
         IF (DENS(1).NE.0 .OR. DENS(NODE).NE.0) THEN
            CALL E1STR (1, DENS(1))
            CALL E1STR (2, DENS(NODE))
            CALL E1STI (1, NODE)
            CALL E1MES (5, 7, 'DENS(1) = %(R1) and DENS(NODE=%(I1)) '//
     &                  '= %(R2).  The beginning and ending initial '//
     &                  'estimates of the density must be zero.')
         END IF
         IF (DENS(ISMIN(NODE,DENS,1)) .LT. 0) THEN
            CALL E1MES (5, 8, 'The initial estimates of the '//
     &                  'density, DENS, must be greater than or '//
     &                  'equal to 0.')
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
         END IF
      END IF
      NOB1 = 0
      DO 10  I=1, NOBS
         IF (X(I).LT.BNDS(1) .OR. X(I).GT.BNDS(2)) THEN
            NOB1 = NOB1 + 1
         END IF
   10 CONTINUE
      IF (NOB1 .EQ. NOBS) THEN
         CALL E1MES (5, 9, 'All elements in X lie outside the '//
     &                  'interval BNDS(1) to BNDS(2).  At least one '//
     &                  'element of X must lie in this interval.')
      END IF
      IF (EPS .LE. 0.0) THEN
         EPS1 = 1.0E-4
      ELSE
         EPS1 = EPS
      END IF
      IF (N1RCD(0) .NE. 0) GO TO 9000
C                                  Initialization
      IMPTR = 0
C                                  Set initial densities
      IF (INIT .EQ. 0) THEN
         DENS(1) = 0.0
         DENS(2) = 2.0/(BNDS(2)-BNDS(1))
         DENS(3) = 0.0
         M = 3
      ELSE
         M = NODE
      END IF
C                                  Refine mesh
   20 IF (INIT .EQ. 0) THEN
         MOLD = M
         IMPTR = IMPTR + 1
         M = MIN0(NODE,MINCR(IMPTR))
      END IF
C                                  Get mesh interval width
      H = (BNDS(2)-BNDS(1))/(M-1)
      H2 = H*H
      H3 = H2*H
C                                  Make initial DENS integrate to 1.
      IF (INIT .NE. 0) THEN
         CALL SSCAL (NODE, 1.0/(H*SSUM(NODE,DENS,1)), DENS, 1)
      END IF
C                                  Set mesh nodes
      B(1) = BNDS(1)
      DO 30  I=2, M
         B(I) = B(I-1) + H
   30 CONTINUE
C                                  Set B indices for interpolating X
      IPTR = 0
   40 IPTR = IPTR + 1
      IF (X(IPTR) .LT. BNDS(1)) GO TO 40
      DO 60  K=1, M - 1
         ILOHI(K,1) = IPTR
         ILOHI(K,2) = IPTR - 1
         IF (IPTR .LE. NOBS) THEN
   50       IF (X(IPTR) .LT. B(K+1)) THEN
               ILOHI(K,2) = ILOHI(K,2) + 1
               IPTR = IPTR + 1
               IF (IPTR .LE. NOBS) GO TO 50
            END IF
         END IF
   60 CONTINUE
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
   70 FACTOR = 2.0*ALPHA/H3
C                                      Initialize mesh node densities
      IF (INIT .EQ. 0) THEN
C                                      Via DESPT
         CALL D2SPT (M-2, B(2), 1, MOLD, BNDS, DENS, DENEST, WK, WK,
     &               WK)
         TEMP = 1.0/(M*M*H)
         DO 80  I=2, M - 1
            DENS(I) = AMAX1(TEMP,SQRT(DENEST(I-1,1)))
   80    CONTINUE
      ELSE
C                                      Via the initial estimates
         DO 90  I=2, M - 1
            DENS(I) = SQRT(DENS(I))
   90    CONTINUE
      END IF
      DENS(M) = 0.0
C                                      Maximize
      DO 140  ITER=1, MAXIT
C                                      Get Hessian - Lagrangian
         HESS(1,1) = 0.0
         HESS(1,2) = 0.0
         HESS(2,1) = 0.0
         BSMALL = 0.0
         SUM = 0.0
C                                      CK** are true estimates = FK**2
         DO 120  K=2, M - 1
            KM1 = K - 1
            KM2 = MAX0(1,K-2)
            KP1 = K + 1
            KP2 = MIN0(M,K+2)
            FK = DENS(K)
            FKM1 = DENS(KM1)
            FKM2 = DENS(KM2)
            CKM2 = FKM2**2
            CKM1 = FKM1**2
            CK = FK**2
            CKP1 = DENS(KP1)**2
            CKP2 = DENS(KP2)**2
            BK = B(K)
            BKM1 = B(KM1)
            SUM = SUM + CK
            IF (K .GE. 4) HESS(1,KM1) = 4.0*FK*FKM2*FACTOR
            SUM1 = 0.0D0
            SUM2 = 0.0D0
            SUM3 = 0.0D0
            DO 100  I=ILOHI(K,1), ILOHI(K,2)
               TEMP = (X(I)-BK)/H
               CONS = (1.0-TEMP)/(CK+(CKP1-CK)*TEMP)
               SUM1 = SUM1 - CONS
               SUM2 = SUM2 + CONS*CONS
  100       CONTINUE
            CKMCM1 = CK - CKM1
            DO 110  I=ILOHI(KM1,1), ILOHI(KM1,2)
               CONS = (X(I)-BKM1)/H
               TEMP = CKM1 + CKMCM1*CONS
               SUM1 = SUM1 - CONS/TEMP
               TEMP = TEMP*TEMP
               SUM2 = SUM2 + (CONS*CONS)/TEMP
               SUM3 = SUM3 + CONS*(1.0-CONS)/TEMP
  110       CONTINUE
            TEMP = FACTOR*(CKM2+CKP2-4.0*(CKM1+CKP1)+6.0*CK) + SUM1
            TEMP = 2.0*TEMP
            BSMALL = BSMALL + 2.0*CK*TEMP
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
               HESS(3,KM1) = TEMP + 4.0*CK*(6.0*FACTOR+SUM2)
               IF (K .NE. 2) HESS(2,KM1) = 4.0*FK*FKM1*(-4.0*FACTOR+SUM3)
               DENEST(KM1,1) = FK*TEMP
               DENEST(KM1,2) = -2.0*FK
  120       CONTINUE
            BSMALL = 1.0/H - SUM + BSMALL
C                                     Save portion of DENEST
            CALL SCOPY (M-2, DENEST(1,2), 1, DENEST(1,3), 1)
C                                     Finish with the hessian
            CALL SADD (M-2, -BSMALL/(2.0*SUM), HESS(3,1), LDHESS)
C                                     Fill out symmetric band structure
            CALL SCOPY (M-4, HESS(1,3), LDHESS, HESS(5,1), LDHESS)
            HESS(5,M-3) = 0.0
            HESS(5,M-2) = 0.0
            CALL SCOPY (M-3, HESS(2,2), LDHESS, HESS(4,1), LDHESS)
            HESS(4,M-2) = 0.0
C                                     Solve symmetric band linear system
            CALL L2TRB (M-2, HESS, LDHESS, 2, 2, HESS, LDHESS, IPVT, WK2)
            CALL LFSRB (M-2, HESS, LDHESS, 2, 2, IPVT, DENEST, 1, DENEST)
            CALL LFSRB (M-2, HESS, LDHESS, 2, 2, IPVT, DENEST(1,2), 1,
     &                  DENEST(1,2))
            IF (N1RCD(1) .NE. 0) GO TO 9000
C                                     Compute the constant
            CONS = SDOT(M-2,DENEST(1,3),1,DENEST(1,2),1)
            CONS = (1.0/H-SUM-SDOT(M-2,DENEST(1,3),1,DENEST(1,1),1))/CONS
C                                     Update the gradient
            CALL SAXPY (M-2, CONS, DENEST(1,2), 1, DENEST(1,1), 1)
C                                     Parameter updates
            CALL SAXPY (M-2, -1.0, DENEST(1,1), 1, DENS(2), 1)
C                                     Check the convergence criterion
            TEMP = SNRM2(M-2,DENS(2),1)
            IF (SNRM2(M-2,DENEST,1) .LT. EPS1*TEMP) GO TO 150
C                                     Ad hoc projection to plus quadrant
            TEMP = TEMP*1.0E-4/SQRT(M-2.0)
            DO 130  I=2, M - 1
               DENS(I) = AMAX1(TEMP,DENS(I))
  130       CONTINUE
  140    CONTINUE
         CALL E1STI (1, MAXIT)
         CALL E1MES (3, 1, 'The maximum number of iterations '//
     &               '(MAXIT=%(I1)) was exceeded.')
C                                     Replace DENS(*) with squares
  150    CALL SHPROD (M-2, DENS(2), 1, DENS(2), 1, DENS(2), 1)
         IF (M .NE. NODE) GO TO 20
C                                     Evaluate log likelihood and penalty
         SUM1 = 0.0
C                                     Penalty
         DO 160  K=1, M
            KM1 = MAX0(K-1,1)
            KP1 = MIN0(K+1,M)
            SUM1 = SUM1 + (DENS(KM1)-2.0*DENS(K)+DENS(KP1))**2
  160    CONTINUE
         STAT(2) = -0.5*FACTOR*SUM1
         SUM2 = 0.0
C                                     Log-likelihood
         DO 170  I=1, NOBS
            IF (X(I).GE.BNDS(1) .AND. X(I).LE.BNDS(2)) THEN
               CALL D2SPT (1, X(I), 1, NODE, BNDS, DENS, DENEST, WK, WK,
     &                     WK)
               SUM2 = SUM2 + ALOG(DENEST(1,1))
            END IF
  170    CONTINUE
         STAT(1) = SUM2
C                                     Evaluate M.L.P.E. mean and variance
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUM1 = 0.0
      SUM2 = 0.0
      DO 180  K=1, M - 1
         FK = DENS(K)
         FKP1 = DENS(K+1)
         BK = B(K)
         CONS = FK + FKP1
         TEMP = CONS + FKP1
         SUM1 = SUM1 + H2*TEMP/6.0 + 0.5*H*BK*CONS
         SUM2 = SUM2 + H3*(TEMP+FKP1)/12.0 + H2*BK*TEMP/3.0 +
     &          0.5*H*BK*BK*CONS
  180 CONTINUE
      STAT(3) = SUM1
      STAT(4) = SUM2 - SUM1*SUM1
C                                    Exit section
 9000 CALL E1POP ('D3SPL ')
      RETURN
      END
/EOF
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
BBBBBBBBBB   1111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111   BBBBBBBB
BBBBBBBBBB   1111111111111111111111111111  Digital Equipment Corporation - VAX/VMS Version V4.7  1111111111111111111111111111   BBBBBBBB
BBBBBBBBBB   1111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111   BBBBBBBB
```

DECNET

NORMAL

.INP;41

ORIGINAL PAGE IS
OF POOR QUALITY

```
File _DUA0:[]NORMAL.INP;41 (252,111,0), last revised on 22-DEC-1988 13:01, is a 1 block sequential file owned by UIC [DECNET].  The
records are variable length with implied (CR) carriage control.  The longest record is 39 bytes.

Job NORMAL (129) queued to TERM$LA120A on 22-DEC-1988 13:01 by user DECNET, UIC [DECNET], under account DECNET at priority 100,
started on printer LTA4: on 22-DEC-1988 13:03 from queue TERM$LA120A.

BBBBBBBBBB   1111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111   BBBBBBBB
BBBBBBBBBB   1111111111111111111111111111  Digital Equipment Corporation - VAX/VMS Version V4.7  1111111111111111111111111111   BBBBBBBB
BBBBBBBBBB   1111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111   BBBBBBBB
```

```
              1          40
     900.0000     45.0000
       8.0000      0.8000
     500.0000     25.0000
       7.0000      0.7000
     250.0000     12.5000
      20.0000      1.0000
     150.0000      7.5000
       0.5000      0.0150
    1500.0000     75.0000
      20.0000      0.6000
     950.0000     25.5000
  21     0        20.00      1.0E-05   30
   2
```

ORIGINAL PAGE IS
OF POOR QUALITY

Digital Equipment Corporation - VAX/VMS Version V4.7

ORIGINAL PAGE IS OF POOR QUALITY

NETNONPRIV

RANDM33

CPR;1

File _DBA0:[]RANDM33.CPR;1 (383,943,0), last revised on 23-NOV-1988 11:26, is a 33 block sequential file owned by UIC [11,11]. The records are variable length with FORTRAN (FTN) carriage control. The longest record is 120 bytes.

Job RANDM33 (689) queued to SYS$SBPRT on 23-NOV-1988 11:27 by user NETNONPRIV, UIC [11,11], under account 20100ADD at priority 100, started on printer _TTF7: on 23-NOV-1988 11:27 from queue TTF7.

Digital Equipment Corporation - VAX/VMS Version V4.7

```
1   40
 0. 9000E+03   0. 4500E+02
LOGNORMAL SF
 0. 8165E+03   0. 9306E+03   0. 8813E+03   0. 8726E+03   0. 9717E+03
 0. 8942E+03   0. 8963E+03   0. 8590E+03   0. 8760E+03   0. 8505E+03
 0. 9148E+03   0. 9603E+03   0. 9484E+03   0. 8751E+03   0. 9161E+03
 0. 8845E+03   0. 8782E+03   0. 9993E+03   0. 9872E+03   0. 8657E+03
 0. 1010E+04   0. 8801E+03   0. 9605E+03   0. 8738E+03   0. 9339E+03
 0. 8546E+03   0. 8574E+03   0. 9446E+03   0. 9070E+03   0. 9341E+03
 0. 8816E+03   0. 9039E+03   0. 8584E+03   0. 8706E+03   0. 8554E+03
 0. 9232E+03   0. 9712E+03   0. 8188E+03   0. 9656E+03   0. 9446E+03
1   40
 0. 8000E+01   0. 8000E+00
LOGNORMAL XLNMF
 0. 6570E+01   0. 8531E+01   0. 7652E+01   0. 7503E+01   0. 9299E+01
 0. 7878E+01   0. 7916E+01   0. 7271E+01   0. 7560E+01   0. 7128E+01
 0. 8245E+01   0. 9083E+01   0. 8859E+01   0. 7546E+01   0. 8268E+01
 0. 7708E+01   0. 7599E+01   0. 7967E+01   0. 9599E+01   0. 7384E+01
 0. 1005E+02   0. 7632E+01   0. 9088E+01   0. 7524E+01   0. 8591E+01
 0. 7197E+01   0. 7243E+01   0. 8789E+01   0. 8104E+01   0. 8596E+01
 0. 7658E+01   0. 8049E+01   0. 7261E+01   0. 7469E+01   0. 7209E+01
 0. 8397E+01   0. 9290E+01   0. 6608E+01   0. 9164E+01   0. 8790E+01
1   40
 0. 5000E+03   0. 2500E+02
LOGNORMAL SO
 0. 4536E+03   0. 5170E+03   0. 4896E+03   0. 4848E+03   0. 5398E+03
 0. 4968E+03   0. 4980E+03   0. 4772E+03   0. 4866E+03   0. 4725E+03
 0. 5082E+03   0. 5335E+03   0. 5269E+03   0. 4862E+03   0. 5090E+03
 0. 4914E+03   0. 4879E+03   0. 4996E+03   0. 5485E+03   0. 4809E+03
 0. 5613E+03   0. 4890E+03   0. 5336E+03   0. 4855E+03   0. 5198E+03
 0. 4748E+03   0. 4763E+03   0. 5248E+03   0. 5039E+03   0. 5190E+03
 0. 4898E+03   0. 5022E+03   0. 4769E+03   0. 4837E+03   0. 4752E+03
 0. 5129E+03   0. 5396E+03   0. 4549E+03   0. 5365E+03   0. 5248E+03
1   40
 0. 7000E+01   0. 7000E+00
LOGNORMAL XLNMO
 0. 5749E+01   0. 7465E+01   0. 6696E+01   0. 6565E+01   0. 8136E+01
 0. 6894E+01   0. 6926E+01   0. 6362E+01   0. 6615E+01   0. 6237E+01
 0. 7214E+01   0. 7947E+01   0. 7752E+01   0. 6602E+01   0. 7235E+01
 0. 6744E+01   0. 6649E+01   0. 6971E+01   0. 8399E+01   0. 6461E+01
 0. 8797E+01   0. 6678E+01   0. 7952E+01   0. 6583E+01   0. 7517E+01
 0. 6297E+01   0. 6338E+01   0. 7690E+01   0. 7091E+01   0. 7521E+01
 0. 6701E+01   0. 7043E+01   0. 6353E+01   0. 6535E+01   0. 6308E+01
 0. 7347E+01   0. 8129E+01   0. 5782E+01   0. 8036E+01   0. 7691E+01
1   40
 0. 2500E+03   0. 1250E+02
LOGNORMAL S
 0. 2268E+03   0. 2585E+03   0. 2448E+03   0. 2424E+03   0. 2699E+03
 0. 2484E+03   0. 2490E+03   0. 2386E+03   0. 2433E+03   0. 2363E+03
 0. 2541E+03   0. 2667E+03   0. 2634E+03   0. 2431E+03   0. 2545E+03
 0. 2457E+03   0. 2439E+03   0. 2498E+03   0. 2742E+03   0. 2405E+03
 0. 2807E+03   0. 2445E+03   0. 2668E+03   0. 2427E+03   0. 2594E+03
 0. 2374E+03   0. 2382E+03   0. 2624E+03   0. 2519E+03   0. 2595E+03
 0. 2449E+03   0. 2511E+03   0. 2385E+03   0. 2418E+03   0. 2376E+03
 0. 2565E+03   0. 2698E+03   0. 2275E+03   0. 2682E+03   0. 2624E+03
1   40
 0. 2000E+02   0. 1000E+01
LOGNORMAL SIGO
-0. 1814E+02  -0. 2068E+02  -0. 1958E+02  -0. 1939E+02  -0. 2159E+02
-0. 1987E+02  -0. 1992E+02  -0. 1909E+02  -0. 1947E+02  -0. 1890E+02
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
-0.2033E+02 -0.2134E+02 -0.2107E+02 -0.1945E+02 -0.2036E+02
-0.1965E+02 -0.1952E+02 -0.1998E+02 -0.2194E+02 -0.1924E+02
-0.2245E+02 -0.1956E+02 -0.2135E+02 -0.1942E+02 -0.2075E+02
-0.1899E+02 -0.1905E+02 -0.2099E+02 -0.2015E+02 -0.2076E+02
-0.1959E+02 -0.2009E+02 -0.1908E+02 -0.1935E+02 -0.1901E+02
-0.2052E+02 -0.2158E+02 -0.1820E+02 -0.2146E+02 -0.2099E+02
1  40
 0.1500E+03  0.7500E+01
LOGNORMAL SIG
 0.1361E+03  0.1551E+03  0.1469E+03  0.1454E+03  0.1619E+03
 0.1490E+03  0.1494E+03  0.1432E+03  0.1460E+03  0.1418E+03
 0.1525E+03  0.1600E+03  0.1581E+03  0.1459E+03  0.1527E+03
 0.1474E+03  0.1464E+03  0.1499E+03  0.1645E+03  0.1443E+03
 0.1684E+03  0.1467E+03  0.1601E+03  0.1456E+03  0.1556E+03
 0.1424E+03  0.1429E+03  0.1574E+03  0.1512E+03  0.1557E+03
 0.1469E+03  0.1507E+03  0.1431E+03  0.1451E+03  0.1426E+03
 0.1539E+03  0.1619E+03  0.1365E+03  0.1609E+03  0.1574E+03
1  40
 0.5000E+00  0.1500E-01
NORMAL XXN
 0.4352E+00  0.4832E+00  0.5104E+00  0.4984E+00  0.5012E+00
 0.4884E+00  0.4749E+00  0.5070E+00  0.5070E+00  0.5227E+00
 0.4956E+00  0.5007E+00  0.5144E+00  0.4727E+00  0.4758E+00
 0.5011E+00  0.5066E+00  0.4637E+00  0.4956E+00  0.4775E+00
 0.4969E+00  0.5073E+00  0.5034E+00  0.5222E+00  0.5153E+00
 0.5010E+00  0.4801E+00  0.5059E+00  0.4968E+00  0.5079E+00
 0.5201E+00  0.5107E+00  0.4905E+00  0.4750E+00  0.5095E+00
 0.4933E+00  0.5051E+00  0.5104E+00  0.5355E+00  0.4948E+00
1  40
NORMAL XXM
 0.4352E+00  0.4832E+00  0.5104E+00  0.4984E+00  0.5012E+00
 0.4884E+00  0.4749E+00  0.5070E+00  0.5070E+00  0.5227E+00
 0.4956E+00  0.5007E+00  0.5144E+00  0.4727E+00  0.4758E+00
 0.5011E+00  0.5066E+00  0.4637E+00  0.4956E+00  0.4775E+00
 0.4969E+00  0.5073E+00  0.5034E+00  0.5222E+00  0.5153E+00
 0.5010E+00  0.4801E+00  0.5059E+00  0.4968E+00  0.5079E+00
 0.5201E+00  0.5107E+00  0.4905E+00  0.4750E+00  0.5095E+00
 0.4933E+00  0.5051E+00  0.5104E+00  0.5355E+00  0.4948E+00
1  40
NORMAL XXG
 0.4352E+00  0.4832E+00  0.5104E+00  0.4984E+00  0.5012E+00
 0.4884E+00  0.4749E+00  0.5070E+00  0.5070E+00  0.5227E+00
 0.4956E+00  0.5007E+00  0.5144E+00  0.4727E+00  0.4758E+00
 0.5011E+00  0.5066E+00  0.4637E+00  0.4956E+00  0.4775E+00
 0.4969E+00  0.5073E+00  0.5034E+00  0.5222E+00  0.5153E+00
 0.5010E+00  0.4801E+00  0.5059E+00  0.4968E+00  0.5079E+00
 0.5201E+00  0.5107E+00  0.4905E+00  0.4750E+00  0.5095E+00
 0.4933E+00  0.5051E+00  0.5104E+00  0.5355E+00  0.4948E+00
1  40
 0.1500E+04  0.7500E+02
NORMAL TF
 0.1176E+04  0.1416E+04  0.1552E+04  0.1492E+04  0.1506E+04
 0.1442E+04  0.1374E+04  0.1535E+04  0.1535E+04  0.1613E+04
 0.1478E+04  0.1504E+04  0.1572E+04  0.1364E+04  0.1379E+04
 0.1506E+04  0.1533E+04  0.1318E+04  0.1478E+04  0.1388E+04
 0.1484E+04  0.1537E+04  0.1517E+04  0.1611E+04  0.1577E+04
 0.1505E+04  0.1400E+04  0.1530E+04  0.1484E+04  0.1540E+04
 0.1601E+04  0.1554E+04  0.1452E+04  0.1375E+04  0.1547E+04
 0.1467E+04  0.1525E+04  0.1552E+04  0.1678E+04  0.1474E+04
1  40
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
  0.2000E+02  0.6000E+00
NORMAL TO
  0.1741E+02  0.1933E+02  0.2042E+02  0.1994E+02  0.2005E+02
  0.1953E+02  0.1900E+02  0.2028E+02  0.2028E+02  0.2091E+02
  0.1982E+02  0.2003E+02  0.2057E+02  0.1891E+02  0.1903E+02
  0.2004E+02  0.2027E+02  0.1855E+02  0.1982E+02  0.1910E+02
  0.1988E+02  0.2029E+02  0.2013E+02  0.2089E+02  0.2061E+02
  0.2004E+02  0.1920E+02  0.2024E+02  0.1987E+02  0.2032E+02
  0.2081E+02  0.2043E+02  0.1962E+02  0.1900E+02  0.2038E+02
  0.1973E+02  0.2020E+02  0.2042E+02  0.2142E+02  0.1979E+02
1  40
  0.8500E+03  0.2550E+02
NORMAL T
  0.7399E+03  0.8215E+03  0.8677E+03  0.8474E+03  0.8521E+03
  0.9302E+03  0.8073E+03  0.8619E+03  0.9619E+03  0.8885E+03
  0.8424E+03  0.8512E+03  0.8744E+03  0.8037E+03  0.8089E+03
  0.8519E+03  0.8613E+03  0.7882E+03  0.8424E+03  0.8118E+03
  0.8447E+03  0.8624E+03  0.8557E+03  0.8877E+03  0.8760E+03
  0.8517E+03  0.8161E+03  0.8601E+03  0.8446E+03  0.8635E+03
  0.8842E+03  0.8682E+03  0.8338E+03  0.8074E+03  0.8661E+03
  0.8387E+03  0.8586E+03  0.8677E+03  0.9104E+03  0.8412E+03
LOG OF CYCLES TO REACH MEAN FATIGUE STR =
50 MPA
  0.6026E+01  0.7799E+01  0.6977E+01  0.6849E+01  0.8486E+01
  0.7199E+01  0.7242E+01  0.6632E+01  0.6896E+01  0.6491E+01
  0.7529E+01  0.8289E+01  0.8074E+01  0.6904E+01  0.7564E+01
  0.7034E+01  0.6931E+01  0.7295E+01  0.8765E+01  0.6754E+01
  0.9179E+01  0.6961E+01  0.8292E+01  0.6851E+01  0.7828E+01
  0.6568E+01  0.6623E+01  0.8017E+01  0.7399E+01  0.7839E+01
  0.6975E+01  0.7339E+01  0.6634E+01  0.6833E+01  0.6574E+01
  0.7669E+01  0.8475E+01  0.6025E+01  0.8351E+01  0.8027E+01
SORTED LOG OF CYCLES
  0.6025E+01  0.6026E+01  0.6491E+01  0.6568E+01  0.6574E+01
  0.6623E+01  0.6632E+01  0.6634E+01  0.6754E+01  0.6833E+01
  0.6849E+01  0.6851E+01  0.6896E+01  0.6904E+01  0.6931E+01
  0.6961E+01  0.6975E+01  0.6977E+01  0.7034E+01  0.7199E+01
  0.7242E+01  0.7295E+01  0.7339E+01  0.7399E+01  0.7529E+01
  0.7564E+01  0.7669E+01  0.7799E+01  0.7828E+01  0.7839E+01
  0.8017E+01  0.8027E+01  0.8074E+01  0.8289E+01  0.8292E+01
  0.8351E+01  0.8475E+01  0.8486E+01  0.8765E+01  0.9179E+01
DESPL PARAMETERS
21    0     0.2000E+02  0.1000E-04   30
BNDS(1),BNDS(2)=   0.5723E+01    0.9638E+01
PDF OF LOG OF CURRENT CYCLES,LOG XNM,Y AXIS OF PDF PLOT
  0.0000E+00  0.3553E-01  0.9400E-01  0.1649E+00  0.2402E+00
  0.3114E+00  0.3720E+00  0.4176E+00  0.4468E+00  0.4591E+00
  0.4550E+00  0.4355E+00  0.4018E+00  0.3560E+00  0.3007E+00
  0.2394E+00  0.1765E+00  0.1164E+00  0.6368E-01  0.2296E-01
  0.0000E+00
OUTPUT STATISTICS
-0.4547E+02 -0.1308E+02  0.7603E+01  0.5956E+00
NUMBER OF MISSING VALUES
  0
LOG OF CURRENT CYCLES, LOG XNM
  0.5723E+01  0.9638E+01  0.5919E+01  0.6115E+01  0.6311E+01
  0.6506E+01  0.6702E+01  0.6898E+01  0.7094E+01  0.7289E+01
  0.7485E+01  0.7681E+01  0.7876E+01  0.8072E+01  0.8268E+01
  0.8464E+01  0.8659E+01  0.8855E+01  0.9051E+01  0.9246E+01
  0.9442E+01
ORDERED LOG OF CURRENT CYCLES, LOG,XNM,              X AXIS PDF, CDF PLOT
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
 0.5723E+01  0.5919E+01  0.6115E+01  0.6311E+01  0.6506E+01
 0.6702E+01  0.6898E+01  0.7094E+01  0.7289E+01  0.7485E+01
 0.7681E+01  0.7876E+01  0.8072E+01  0.8268E+01  0.8464E+01
 0.8659E+01  0.8855E+01  0.9051E+01  0.9246E+01  0.9442E+01
 0.9638E+01
GCDF PARAMETERS
 2
CDF OF LOG OF CURRENT CYCLES, LOG XNM,                 Y AXIS OF PDF, CDF PLOT
 0.0000E+00  0.3477E-02  0.1615E-01  0.4149E-01  0.8113E-01
 0.1351E+00  0.2020E+00  0.2793E+00  0.3638E+00  0.4525E+00
 0.5419E+00  0.6291E+00  0.7110E+00  0.7852E+00  0.8494E+00
 0.9023E+00  0.9430E+00  0.9716E+00  0.9893E+00  0.9978E+00
 0.1000E+01
ORDERED LOG OF CURRENT CYCLES, LOG XNM,                X AXIS OF PDF, CDF PLOT
 0.5723E+01  0.5919E+01  0.6115E+01  0.6311E+01  0.6506E+01
 0.6702E+01  0.6898E+01  0.7094E+01  0.7289E+01  0.7485E+01
 0.7681E+01  0.7876E+01  0.8072E+01  0.8268E+01  0.8464E+01
 0.8659E+01  0.8855E+01  0.9051E+01  0.9246E+01  0.9442E+01
 0.9638E+01
 0.5723E+01  0.5919E+01  0.6115E+01  0.6311E+01  0.6506E+01
 0.6702E+01  0.6898E+01  0.7094E+01  0.7289E+01  0.7485E+01
 0.7681E+01  0.7876E+01  0.8072E+01  0.8268E+01  0.8464E+01
 0.8659E+01  0.8855E+01  0.9051E+01  0.9246E+01  0.9442E+01
 0.9638E+01
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
BBBBBBBBBB  666666666666666666666666666666666666666666666666666666666666666666666666666666666666666666666666666666  BBBBBBBBBB
BBBBBBBBBB  666666666666666666666666  Digital Equipment Corporation - VAX/VMS Version V4.7  666666666666666666666666  BBBBBBBBBB
BBBBBBBBBB  666666666666666666666666666666666666666666666666666666666666666666666666666666666666666666666666666666  BBBBBBBBBB
```

NETNONPRIV

PLOT1

CPR;1

```
File _DBA0:[]PLOT1.CPR;1 (359,209,0), last revised on 23-NOV-1988 11:26, is a 2 block sequential file owned by UIC [11,11].  The
records are variable length with FORTRAN (FTN) carriage control.  The longest record is 25 bytes.

Job PLOT1 (687) queued to SYS$SBPRT on 23-NOV-1988 11:26 by user NETNONPRIV, UIC [11,11], under account 20100ADD at priority 100,
started on printer _TTF7: on 23-NOV-1988 11:26 from queue TTF7.

BBBBBBBBBB  666666666666666666666666666666666666666666666666666666666666666666666666666666666666666666666666666666  BBBBBBBBBB
BBBBBBBBBB  666666666666666666666666  Digital Equipment Corporation - VAX/VMS Version V4.7  666666666666666666666666  BBBBBBBBBB
BBBBBBBBBB  666666666666666666666666666666666666666666666666666666666666666666666666666666666666666666666666666666  BBBBBBBBBB
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
(E12.4,1X,E12.4)
 0.5723E+01    0.0000E+00
 0.5919E+01    0.3553E-01
 0.6115E+01    0.9400E-01
 0.6311E+01    0.1649E+00
 0.6506E+01    0.2402E+00
 0.6702E+01    0.3114E+00
 0.6898E+01    0.3720E+00
 0.7094E+01    0.4176E+00
 0.7289E+01    0.4468E+00
 0.7485E+01    0.4591E+00
 0.7681E+01    0.4550E+00
 0.7876E+01    0.4355E+00
 0.8072E+01    0.4018E+00
 0.8268E+01    0.3560E+00
 0.8464E+01    0.3007E+00
 0.8659E+01    0.2394E+00
 0.8855E+01    0.1765E+00
 0.9051E+01    0.1164E+00
 0.9246E+01    0.6368E-01
 0.9442E+01    0.2296E-01
 0.9638E+01    0.0000E+00
```

```
LLLLLLLLLL   3333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333   LLLLLLLLLL
LLLLLLLLLL   333333333333333333333333  Digital Equipment Corporation - VAX/VMS Version V4.7  333333333333333333333333   LLLLLLLLLL
LLLLLLLLLL   3333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333   LLLLLLLLLL
```

NETNONPRIV

PLOT2

CPR;1

ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

File _DBA0:[]PLOT2.CPR;1 (363,178,0), last revised on 23-NOV-1988 11:26, is a 2 block sequential file owned by UIC [11,11]. The records are variable length with FORTRAN (FTN) carriage control. The longest record is 25 bytes.

Job PLOT2 (688) queued to SYS$$PRT on 23-NOV-1988 11:26 by user NETNONPRIV, UIC [11,11], under account 20100ADD at priority 100, started on printer _TTF6: on 23-NOV-1988 11:26 from queue TTF6.


```
LLLLLLLLLL   3333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333   LLLLLLLLLL
LLLLLLLLLL   333333333333333333333333  Digital Equipment Corporation - VAX/VMS Version V4.7  333333333333333333333333   LLLLLLLLLL
LLLLLLLLLL   3333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333333   LLLLLLLLLL
```

```
(E12.4,1X,E12.4)
 0.5723E+01   0.0000E+00
 0.5919E+01   0.3477E-02
 0.6115E+01   0.1615E-01
 0.6311E+01   0.4149E-01
 0.6506E+01   0.8113E-01
 0.6702E+01   0.1351E+00
 0.6898E+01   0.2020E+00
 0.7094E+01   0.2793E+00
 0.7289E+01   0.3638E+00
 0.7485E+01   0.4525E+00
 0.7681E+01   0.5419E+00
 0.7876E+01   0.6291E+00
 0.8072E+01   0.7110E+00
 0.8268E+01   0.7852E+00
 0.8464E+01   0.8494E+00
 0.8659E+01   0.9023E+00
 0.8855E+01   0.9430E+00
 0.9051E+01   0.9716E+00
 0.9246E+01   0.9893E+00
 0.9442E+01   0.9978E+00
 0.9638E+01   0.1000E+01
```

8.0 APPENDIX C

RANDOM4 SAMPLE PROBLEM: SOURCE, INPUT AND OUTPUTFILES

```
DDDDDDDDDD   33333333333333333333333333333333333333333333333333333333333333333333333333333333333   DDDDDDDD
DDDDDDDDDD   333333333333333333333333  Digital Equipment Corporation - VAX/VMS Version V4.7  333333333333333333333333   DDDDDDDD
DDDDDDDDDD   33333333333333333333333333333333333333333333333333333333333333333333333333333333333   DDDDDDDD
```

DECNET

NR4

FOR;5


```
File _DUAO:[]NR4.FOR;5 (431,64,0), last revised on 22-DEC-1988 13:22, is a 75 block sequential file owned by UIC [DECNET].  The
records are variable length with implied (CR) carriage control.  The longest record is 72 bytes.

Job NR4 (131) queued to TERM$LA120A on 22-DEC-1988 13:25 by user DECNET, UIC [DECNET], under account DECNET at priority 100, started
on printer LTA4: on 22-DEC-1988 13:25 from queue TERM$LA120A.

DDDDDDDDDD   33333333333333333333333333333333333333333333333333333333333333333333333333333333333   DDDDDDDD
DDDDDDDDDD   333333333333333333333333  Digital Equipment Corporation - VAX/VMS Version V4.7  333333333333333333333333   DDDDDDDD
DDDDDDDDDD   33333333333333333333333333333333333333333333333333333333333333333333333333333333333   DDDDDDDD
```

```
JOB,JN=COMP4,US=USAQ530,RT=60,T=30,MFL=2000000.
ACCOUNT,UPW=LOLAB.
DELETE,PDN=NR4BLD,ID=SMBOYCE.
CFT77.
REWIND,DN=$BLD.
BUILD,I=0,OBL=0.
SAVE,DN=$NBL,PDN=NR4BLD,ID=SMBOYCE.
DELETE,PDN=NR4BLD,ID=SMBOYCE,ED=-1.
/EOF
C CHAMIS MICROMECHANICS CONSTITUTIVE EQUATIONS;
C RANDOMIZED AND APPLIED TO FATIGUE STRENGTH
      INTEGER NTOT,ISEED,M,INIT,NMISS,MAXIT,NODE
      REAL XM,XS,YM,YS,EPS,P,RWKSP(9999),ALPHA
      DIMENSION SF(10000),XLNMF(10000),SO(10000)
      DIMENSION XLNMO(10000),S(10000)
      DIMENSION SIGO(10000),SIG(10000)
      DIMENSION XXN(10000),XXM(10000),XXQ(10000)
      DIMENSION XNM(10000),BNDSX(10000),DISTX(10000)
      DIMENSION TF(10000),TO(10000),T(10000)
      DIMENSION STAT(9999)
      DIMENSION BNDS(999),DENS(999)
      DIMENSION C(999),PP(999)
      DIMENSION SM(10)
      DIMENSION XP(1),CUM(1)
      DIMENSION AL(12)
C     COMMON/MEP1/KPRINT,TOL,MAXFN
C     KPRINT=1
C     TOL=1.0E-06
C     MAXFN=50
 1001 FORMAT(5E12.4)
 1002 FORMAT(E12.4,2X,I4)
 1003 FORMAT(I4,I4)
 1004 FORMAT(I4)
 1005 FORMAT(I12,I12)
 1006 FORMAT(2E12.4)
C LOGNORMAL ULTIMATE TENSILE STRENGTH, SF
      READ (5,1005) ISEED,NTOT
      WRITE(6,1005) ISEED,NTOT
C       XM=900.
C       XS=45.
      READ(5,1006)XM,XS
      WRITE(6,1006)XM,XS
      YS = SQRT(LOG(1.0+(XS/XM)**2.))
      YM = LOG(XM) - 0.5*YS**2.
      CALL RNSET(ISEED)
      CALL RNLNL(NTOT,YM,YS,SF)
      WRITE(18,1001)(SF(I),I=1,NTOT)
      WRITE(6,2020)
 2020 FORMAT('LOGNORMAL SF')
      WRITE(6,1001)(SF(I),I=1,NTOT)
C LOGNORMAL LOG OF FINAL CYCLE, XLNMF
      WRITE(6,1005) ISEED,NTOT
      READ(5,1006)XM,XS
      WRITE(6,1006)XM,XS
C       XM  = 8.
C       XS  = 0.8
      YS = SQRT( LOG(1.0+(XS/XM)**2) )
      YM = LOG(XM) - 0.5*YS**2
      CALL RNSET(ISEED)
      CALL RNLNL(NTOT,YM,YS,XLNMF)
      WRITE(19,1001) (XLNMF(I),I=1,NTOT)
      WRITE(6,2021)
 2021 FORMAT(' LOGNORMAL XLNMF')
      WRITE(6,1001) (XLNMF(I),I=1,NTOT)
```

```
C LOGNORMAL FATIGUE STRENGTH AT REFERENCE CONDITIONS, SO
      WRITE(6,1005) ISEED,NTOT
      READ(5,1006)XM,XS
      WRITE(6,1006)XM,XS
C       XM  = 500.
C       XS  = 25.
      YS = SQRT( LOG(1.0+(XS/XM)**2) )
      YM = LOG(XM) - 0.5*YS**2
      CALL RNSET(ISEED)
      CALL RNLNL(NTOT,YM,YS,SO)
      WRITE(20,1001) (SO(I),I=1,NTOT)
      WRITE(6,2022)
 2022 FORMAT(' LOGNORMAL SO')
      WRITE( 6,1001) (SO(I),I=1,NTOT)
C LOGNORMAL LOG OF REFERENCE CYCLES, XLNMO
      WRITE(6,1005)ISEED,NTOT
      READ(5,1006)XM,XS
      WRITE(6,1006)XM,XS
C       XM = 7.
C       XS = 0.7
      YS = SQRT(LOG(1.0+(XS/XM)**2))
      YM = LOG(XM) - 0.5*YS**2
      CALL RNSET(ISEED)
      CALL RNLNL(NTOT,YM,YS,XLNMO)
      WRITE (21,1001)(XLNMO(I),I=1,NTOT)
      WRITE (6,2023)
 2023 FORMAT (' LOGNORMAL XLNMO')
      WRITE (6,1001)(XLNMO(I),I=1,NTOT)
C LOGNORMAL FATIGUE STRENGTH AT CURRENT CONDITIONS, S
      WRITE(6,1005)ISEED,NTOT
      READ(5,1006)XM,XS
      WRITE(6,1006)XM,XS
C       XM = 250.
C       XS = 12.5
      YS = SQRT(LOG(1.0+(XS/XM)**2.))
      YM = LOG(XM) - 0.5*YS**2.
      CALL RNSET(ISEED)
      CALL RNLNL(NTOT,YM,YS,S)
      WRITE(22,1001)(S(I),I=1,NTOT)
      WRITE (6,2024)
 2024 FORMAT(' LOGNORMAL S')
      WRITE(6,1001)(S(I),I=1,NTOT)
C DEFINE RANDOM STRESSES
C LOGNORMAL REFERENCE STRESS, SIGO
      WRITE(6,1005)ISEED,NTOT
      READ(5,1006)XM,XS
      WRITE(6,1006)XM,XS
C       XM = 20.
C       XS = 1.
      YS=SQRT(LOG(1.0+(XS/XM)**2.))
      YM=LOG(XM) - 0.5*YS**2.
      CALL RNSET(ISEED)
      CALL RNLNL(NTOT,YM,YS,SIGO)
C CHANGE SIGO TO NEGATIVE VALUES FOR COMPRESSIVE
C RESIDUAL STRESSES
      DO 401 I = 1,NTOT
      SIGO(I)=-SIGO(I)
  401 CONTINUE
      WRITE(26,1001)(SIGO(I),I=1,NTOT)
      WRITE(6,2036)
 2036 FORMAT(' LOGNORMAL SIGO')
      WRITE(6,1001)(SIGO(I),I=1,NTOT)
C LOGNORMAL CURRENT STRESS, SIG
      WRITE(6,1005)ISEED,NTOT
```

```
      READ(5,1006)XM,XS
      WRITE(6,1006)XM,XS
C      XM= 150.
C      XS= 7.5
      YS=SQRT(LOG(1.0+(XS/XM)**2.))
      YM=LOG(XM) - 0.5*YS**2.
      CALL RNSET(ISEED)
      CALL RNLNL(NTOT,YM,YS,SIG)
      WRITE(27,1001)(SIG(I),I=1,NTOT)
      WRITE(6,2037)
 2037 FORMAT(' LOGNORMAL SIG')
      WRITE(6,1001)(SIG(I),I=1,NTOT)
C NORMAL EXPONENTS, XXN,XXM,XXQ
C      YM = 0.5
C      YS = 0.015
      WRITE(6,1005)ISEED,NTOT
      READ(5,1006)YM,YS
      WRITE(6,1006)YM,YS
      CALL RNSET(ISEED)
      CALL RNNOR(NTOT,XXN)
      DO 101 I=1,NTOT
      XXN(I)=YS*XXN(I)+YM
  101 CONTINUE
      WRITE(23,1001)(XXN(I),I=1,NTOT)
      WRITE(6,2025)
 2025 FORMAT(' NORMAL XXN')
      WRITE(6,1001)(XXN(I),I=1,NTOT)
      WRITE(6,1005)ISEED,NTOT
      CALL RNSET(ISEED)
      CALL RNNOR(NTOT,XXM)
      DO 201 I=1,NTOT
      XXM(I)=YS*XXM(I)+YM
  201 CONTINUE
      WRITE(24,1001)(XXM(I),I=1,NTOT)
      WRITE(6,2026)
 2026 FORMAT(' NORMAL XXM')
      WRITE(6,1001)(XXM(I),I=1,NTOT)
      WRITE(6,1005)ISEED,NTOT
      CALL RNSET(ISEED)
      CALL RNNOR(NTOT,XXQ)
      DO 301 I=1,NTOT
      XXQ(I)=YS*XXQ(I)+YM
  301 CONTINUE
      WRITE(25,1001)(XXQ(I),I=1,NTOT)
      WRITE(6,2027)
 2027 FORMAT(' NORMAL XXQ')
      WRITE(6,1001)(XXQ(I),I=1,NTOT)
C DEFINE DETERMINISTIC TEMPERATURES
C      TF=1500.
C      TO=20.
C      T=850.
C NORMAL TEMPERATURES,TF,TO,T
C NORMAL FINAL (MELTING)TEMPERATURE,TF
      WRITE(6,1005)ISEED,NTOT
      READ(5,1006)YM,YS
      WRITE(6,1006)YM,YS
C      YM=1500.
C      YS=75.
      CALL RNSET(ISEED)
      CALL RNNOR(NTOT,TF)
      DO 405 I=1,NTOT
      TF(I)=YS*TF(I)+YM
  405 CONTINUE
      WRITE(6,2046)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
 2046 FORMAT(' NORMAL TF')
      WRITE(6,1001)(TF(I),I=1,NTOT)
C NORMAL REFERENCE TEMPERATURE,TO
      WRITE(6,1005)ISEED,NTOT
      READ(5,1006)YM,YS
      WRITE(6,1006)YM,YS
C      YM=20.
C      YS=0.6
      CALL RNSET(ISEED)
      CALL RNNOR(NTOT,TO)
      DO 406 I=1,NTOT
      TO(I)=YS*TO(I)+YM
  406 CONTINUE
      WRITE(6,2047)
 2047 FORMAT(' NORMAL TO')
      WRITE(6,1001)(TO(I),I=1,NTOT)
C NORMAL CURRENT TEMPERATURE,T
      WRITE(6,1005)ISEED,NTOT
      READ(5,1006)YM,YS
      WRITE(6,1006)YM,YS
C      YM= 850.
C      YS= 42.5
      CALL RNSET(ISEED)
      CALL RNNOR(NTOT,T)
      DO 407 I=1,NTOT
      T(I)=YS*T(I)+YM
  407 CONTINUE
      WRITE(6,2048)
 2048 FORMAT(' NORMAL T')
      WRITE(6,1001)(T(I),I=1,NTOT)
C CALCULATE CURRENT LOG OF CYCLES, LOG XNM
      DO 102 I=1,NTOT
      RS=((SF(I)-SIG(I))/(SF(I)-SIGO(I)))**XXM(I)
C     WRITE(6,6876)RS
C6876 FORMAT(' RS=',E12.4)
C     TEMP=((TF-T)/(TF-TO))**XXN(I)
      TEMP=((TF(I)-T(I))/(TF(I)-TO(I)))**XXN(I)
C     WRITE(6,7876)TEMP
C7876 FORMAT(' TEMP=',E12.4)
C     SS=S(I)
C     SSO=SO(I)
C     XXXQ=XXQ(I)
C     WRITE(6,1001)SS
C     WRITE(6,1001)SSO
C     WRITE(6,1001)XXXQ
      XNM1=(S(I)/(SO(I)*TEMP*RS))**(1./XXQ(I))
C     WRITE(6,8876) XNM1
C8876 FORMAT(' XNM1=',E12.4)
      XNM2=(XLNMF(I)-((XLNMF(I)-XLNMO(I))*XNM1))
C     WRITE(6,8875)XNM2
C8875 FORMAT(' XNM2=',E12.4)
      IF(XNM2.LT.0.0)XNM2=0.0
      XNM(I)=XNM2
C     XNM(I)=10.**XNM2
  102 CONTINUE
      WRITE(28,1001)(XNM(I),I=1,NTOT)
      WRITE(6,2028)
 2028 FORMAT (' LOG OF CYCLES TO REACH MEAN FATIGUE STR =  ',/,
     1'250 MPA')
      WRITE(6,1001)(XNM(I),I=1,NTOT)
C SORT LOG OF CYCLES
      CALL SORT(XNM,NTOT)
      WRITE(29,1001)(XNM(I),I=1,NTOT)
      WRITE (6,2029)
```

ORIGINAL PAGE IS
OF POOR QUALITY

C-2

```
 2029 FORMAT (' SORTED LOG OF CYCLES')
      WRITE (6,1001)(XNM(I),I=1,NTOT)
C CALCULATE PDF OF LOG OF CURRENT CYCLES, LOG XNM
C USING THE MAXIMUM ENTROPY METHOD
C CALCULATE SAMPLE MOMENTS,SM
C NUMBER OF MOMENTS, MMM
      MMM=4
      CALL SMOM(XNM,MMM,NTOT,SM)
      WRITE(30,1001)(SM(I),I=1,MMM)
      WRITE(6,2038)
 2038 FORMAT(' SAMPLE MOMENTS')
      WRITE(6,1001)(SM(I),I=1,MMM)
C OBTAIN MAXIMUM ENTROPY DISTRIBUTION
      KSTART=1
      KDATA=1
C CALCULATE MAX AND MIN ORDINATES FOR PDF (AND CDF)
      BNDS(1) = XNM(1) - 0.05*XNM(1)
      BNDS(2) = XNM(NTOT) + 0.05*XNM(NTOT)
      WRITE (66,8877) BNDS(1),BNDS(2)
      WRITE (6,8877) BNDS(1),BNDS(2)
 8877 FORMAT (' BNDS(1),BNDS(2)=',E12.4,1X,E12.4)
      CALL MEP1(MMM,SM,BNDS(1),BNDS(2),0,XP,KSTART,KDATA,AL,CUM)
      WRITE(31,1001)(AL(I),I=1,MMM+1)
      WRITE(6,2039)
 2039 FORMAT(' LAGRANGIAN MULTIPLIERS')
      WRITE(6,1001)(AL(I),I=1,MMM+1)
C CALCULATE VALUES OF ORDINATES FOR PDF (AND CDF)
C NUMBER OF ORDINATES USED
C
C CALCULATE WINDOW WIDTH, HH
      NODE=21
      HH=(BNDS(2)-BNDS(1))/(NODE-1)
C
C CALCULATE VALUES OF LOG OF CURRENT CYCLES AT WHICH PDF IS ESTIMATED;
C ALSO CALLED 'NODE' VALUES
C
      DO 6001,I=1,NODE-2
      BNDS(I+2)=BNDS(1) + (I*HH)
 6001 CONTINUE
      WRITE(6,983)
  983 FORMAT(' LOG OF CURRENT CYCLES, LOG XNM')
      WRITE(6,1001)(BNDS(I),I=1,NODE)
C
C REORDER BNDS FOR PLOTTING
C
      SAVE1 = BNDS(2)
      SAVE2 = BNDS(NODE)
      BNDS(NODE)=BNDS(2)
      DO 6002, I=1, NODE-2
      BNDS(I+1)=BNDS(I+2)
 6002 CONTINUE
      BNDS(NODE-1)=SAVE2
      BNDS(NODE)=SAVE1
      WRITE(6,984)
  984 FORMAT(' ORDERED LOG OF CURRENT CYCLES, LOG,XNM,
     1X AXIS PDF, CDF PLOT')
      WRITE(6,1001)(BNDS(I),I=1,NODE)
C CALCULATE VALUES OF THE PDF AT EACH ORDINATE
      DO 108 I=1,NODE
C FOR 4 MOMENTS THERE ARE 5 LAGRANGIAN MULTIPLIERS
      DENS(I)=EXP(AL(1)+AL(2)*BNDS(I)+AL(3)*BNDS(I)**2
     1+AL(4)*BNDS(I)**3+AL(5)*BNDS(I)**4)
  108 CONTINUE
C WRITE LOG OF CURRENT CYCLES AND PDF OF LOG OF CURRENT CYCLES,
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C     LOG XNM TO PLOT FILES
      WRITE(34,990)
  990 FORMAT(' (E12.4,1X,E12.4)')
      WRITE(34,991)(BNDS(J),DENS(J),J=1,NODE)
  991 FORMAT(E12.4,1X,E12.4)
C
C CALCULATE CDF OF LOG OF CURRENT CYCLES
      IOPT=2
C
C      READ(3,1004)IOPT
      WRITE(6,992)
  992 FORMAT(' GCDF PARAMETERS')
      WRITE(6,1004)IOPT
      XO=BNDS(1)
      DO 6003,I=1,NODE
      P=GCDF(XO,IOPT,NODE,BNDS,DENS)
      BNDSX(I)=XO
      XO=XO+HH
      DISTX(I)=P
 6003 CONTINUE
      WRITE(6,994)
  994 FORMAT(' CDF OF LOG OF CURRENT CYCLES, LOG XNM,
     1Y AXIS OF PDF, CDF PLOT')
      WRITE(6,1001)(DISTX(I),I=1,NODE)
C
      WRITE(6,993)
  993 FORMAT(' ORDERED LOG OF CURRENT CYCLES, LOG XNM,
     1X AXIS OF PDF, CDF PLOT')
      WRITE(6,1001)(BNDS(I),I=1,NODE)
      WRITE(6,1001)(BNDSX(I),I=1,NODE)
C
C WRITE LOG OF CURRENT CYCLES AND CDF OF LOG OF CURRENT
C TO THE PLOT FILES
C
      WRITE(35,990)
      WRITE(35,991)(BNDS(J),DISTX(J),J=1,NODE)
      STOP
      END
C
      SUBROUTINE SORT (Y,N)
      DIMENSION Y(10000)
C Y IS THE ARRAY TO BE SORTED
C AT COMPLETION Y(1) IS SMALLEST VALUE
C AT COMPLETION Y(N) IS LARGEST VALUE
      N1 = N - 1
      DO 1 I=1,N1
      J = I + 1
      DO 2 K=J,N
      IF (Y(I).LT.Y(K))GO TO 2
      TEMP = Y(I)
      Y(I) = Y(K)
      Y(K) = TEMP
    2 CONTINUE
    1 CONTINUE
      RETURN
      END
C
      SUBROUTINE SMOM(X,M,NSAMP,SM)
C CALCULATES SAMPLE CENTRAL MOMENTS
C X(I) = SAMPLE VALUES, DIMENSION NSAMP
C M = NUMBER OF MOMENTS DESIRED
C NSAMP = SAMPLE SIZE
C SM = VALUE OF MOMENTS, DIMENSION M
      DIMENSION X(10000),SM(10)
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
C CALCULATE MEAN
      SUM=0.0
      DO 1 I=1,NSAMP
    1 SUM=SUM+X(I)
      SM(1)=SUM/FLOAT(NSAMP)
      IF(M.LT.2)RETURN
C CALCULATE VARIANCE
      SUM=0.0
      DO 2 I=1,NSAMP
    2 SUM=SUM+(X(I)-SM(1))**2
      SM(2)=SUM/(FLOAT(NSAMP-1))
      IF(M.LT.3)RETURN
C CALCULATE HIGHER MOMENTS
      DO 4 I=3,M
      SUM=0.0
      DO 3 J=1,NSAMP
    3 SUM=SUM+(X(J)-SM(1))**I
      SM(I)=SUM/FLOAT(NSAMP)
    4 CONTINUE
      RETURN
      END

      SUBROUTINE MEP1(N,CM,XMIN,XMAX,NXP,XP,KSTART,KDATA,AL,CUM)
C     IMPLICIT REAL*8 (A-H,O-Z)
C.... EXECUTIVE PROGRAM FOR USING MAXIMUM ENTROPY METHOD CONSTRAINED BY
C     MOMENTS TO GENERATE A DENSITY FUNCTION
C
      DIMENSION AL(*), CM(*), ETA(4), XP(*),CUM(*),CC(8),ALS(10)
      COMMON /FAIL/ NFAIL
      COMMON/HELP/S(101),XX(16,101),C(8),M
C.... ABOVE LINE DIFFERENT FROM TEXT
C     COMMON/MEP1/KPRINT,TOL,MAXFN
      DATA KPRINT,TOL,MAXFN/1,1.E-6,70/
      IF (N.EQ.1) KSTART=2
C
C     WRITE THE INPUT DATA
C
      IF (KDATA.EQ.0) GO TO 1
      WRITE (6,24)
      WRITE (6,25) KDATA
      WRITE (6,26) KPRINT
      WRITE (6,28) N
      WRITE (6,29) XMAX
      WRITE (6,30) XMIN
      WRITE (6,31) (CM(I),I=1,4)
      IF (N.GT.4) WRITE (6,21) (CM(I),I=5,N)
      IF(ABS(CM(1)).LT.1.E-4)GO TO 48
      WRITE (6,32) TOL
      WRITE (6,33) NXP
    1 CONTINUE
      NFAIL=0
      M=31
      X2MIN=0.0
      X2MAX=1.
C     SAVE CM
      DO 100 I=1,N
  100 CC(I)=CM(I)
C
C     CALCULATE THE MOMENTS AT THE MODIFIED LIMITS
C
      CALL TRN1 (XMAX,XMIN,CC,X2MAX,X2MIN,N)
C
C     CALCULATE THE MOMENTS ABOUT THE ORIGIN FOR THE MODIFIED LIMITS ,
C     STORE THEM IN COMMON IN C
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C
      CALL CONVER(CC,N)
C
C     GENERATE THE SIMPSON MULTIPLIERS AND STORE THEM IN HELP COMMON
C
      CALL SIMSON
C
C     GENERATE THE X,S POWER FOR SUBROUTINE FUNCT, STORE THEM IN HELP
C     COMMON ARRAY
C
      CALL MULTI (X2MAX,X2MIN,N)
C
C     DEFINE THE INPUT DATA FOR SUBROUTINE MPOPT
C
      ETA(1)=1.D-12
      ETA(2)=TOL
      ETA(3)=1.D-24
      ETA(4)=1.D-24
      MODE=1
      UMIN=0.0
C
C     WRITE THE INTERMEDIATE RESULTS YOU HAVE OBTAINED SO FAR
C
      IF (KPRINT.EQ.0) GO TO 2
      WRITE (6,34)
      WRITE (6,35) M
      WRITE (6,36) X2MAX,X2MIN
      WRITE (6,37) (CC(I),I=1,4)
      IF (N.GT.4) WRITE (6,22) (CC(I),I=5,N)
      WRITE (6,38) (C(I),I=1,4)
      IF (N.GT.4) WRITE (6,22) (C(I),I=5,N)
      WRITE (6,39) (ETA(I),I=1,4)
    2 CONTINUE
C
C     FIND A STARTING POINT FOR SUBROUTINE MPOPT TO START THE OPTIMIZAT-
C     ION ALGORITHM
C
      IF(KSTART.EQ.0)GO TO 16
      IF(KSTART.EQ.4)WRITE(6,44)
      CALL START (X2MAX,X2MIN,AL,KSTART,CC,N,KPRINT,UMIN,MODE,MAXFN,ETA)
      IF (NFAIL.EQ.1) GO TO 9
C
C     PRINT THE STARTING VALUES
C
      IF (KPRINT.EQ.0) GO TO 7
      GO TO (3,4,5,6), KSTART
    3 WRITE (6,40)
      WRITE (6,41) (AL(I),I=1,4)
      IF (N.GT.4) WRITE (6,22) (AL(I),I=5,N)
      GO TO 7
    4 WRITE (6,42)
      WRITE (6,41) (AL(I),I=1,4)
      IF (N.GT.4) WRITE (6,22) (AL(I),I=5,N)
      GO TO 7
    5 WRITE (6,43)
      WRITE (6,41) (AL(I),I=1,4)
      IF (N.GT.4) WRITE (6,22) (AL(I),I=5,N)
      GO TO 7
    6 WRITE (6,41) (AL(I),I=1,4)
      IF (N.GT.4) WRITE (6,22) (AL(I),I=5,N)
      GO TO 7
   16 CONTINUE
      RANGE=XMAX-XMIN
C.... CHANGE STARTING VALUES TO 0-1 DOMAIN FOR KSTART=0
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C.... THIS ALGORITHM IS SIMILAR TO TRN2 BUT APPEARS TO GIVE BETTER
C     NUMERICAL RESULTS
      NPL=N+1
      IF(ABS(XMIN).LT.1.E-10)GO TO 19
      DO 17 I=2,NPL
      ALS(I)=0.0
      I1=I-1
      DO 18 J=I1,N
      ALS(I)=ALS(I)+FACTO(J)*XMIN**(J-I1)*RANGE**I1*AL(J+1)/FACTO(I1)
     1/FACTO(J-I1)
18    CONTINUE
17    CONTINUE
      GO TO 50
19    DO 20 I=2,NPL
20    ALS(I)=RANGE**(I-1)*AL(I)
C.... PUT AL(I) IN PROPER LOCATIONS
50    DO 51 I=1,N
51    AL(I)=ALS(I+1)

7     CONTINUE
      NFAIL=0
      IF (KPRINT.EQ.0) GO TO 8
      WRITE (6,45)
8     CONTINUE
      AL(N+1)=2.
      AL(N+2)=0.0
      CALL MPOPT (AL,N,ETA,UMIN,MAXFN,MODE,KPRINT)
      IF (NFAIL.EQ.0) GO TO 10
      IF (KSTART.EQ.4) GO TO 9
C
C     THE PROGRAM HAS FAILED SO FAR , TRY ANOTHER STARTING POINT AND TRY
C     AGAIN
C
      KSTART=KSTART+1
      IF (KSTART.EQ.4.AND.N.LE.2) GO TO 9
      GO TO 2
9     CONTINUE
      WRITE (6,46)
      CALL EXIT
10    CONTINUE
C
C     CALCULATE THE ZEROTH LAGRANGIAN MULTIPLIER
C
      SUM=0.0
      DO 12 I=1,M
      SZ=0.0
      DO 11 K=1,N
      SZ=SZ+AL(K)*XX(K,I)
11    CONTINUE
      SUM=SUM+S(I)*EXP(SZ)
12    CONTINUE
      NPL=N+1
      DO 13 I=1,N
      K=N+2-I
      AL(K)=AL(K-1)
13    CONTINUE
      DELTA=(X2MAX-X2MIN)/FLOAT(M-1)
      AL(1)=-ALOG(SUM*DELTA/3.)
      WRITE(6,101)UMIN
101   FORMAT(26H SUM OF RESIDUALS SQUARED=,E12.5)
      IF (KPRINT.EQ.0) GO TO 14
      WRITE (6,47) (AL(I),I=1,NPL)
14    CONTINUE
C.... RESET KSTART TO ZERO
```

```
      KSTART=0
C
C     CALCULATE THE LAGRANGIAN MULTIPLIERS FOR THE ORIGINAL LIMITS
C
      CALL TRN2 (XMAX,XMIN,AL,X2MAX,X2MIN,N)
C
C     CALCULATE THE CUMULATIVE DISTRIBUTION FUNCTION VALUE AT THE GIVEN
C     POINT
      IF(NXP.EQ.0)RETURN
      DO 15 I=1,NXP
      CUM(I)=CDF(XMIN,XMAX,XP(I),AL,NPL)
   15 CONTINUE
      RETURN
C
   21 FORMAT (57X,4E18.9,//)
   22 FORMAT (57X,4E18.9,//)
   24 FORMAT (1H1,//,20X,'INPUT DATA FOR SUBROUTINE   MEP1',/,20X,33('-'
     1),//)
   25 FORMAT (' INPUT DATA IS PRINTED OUT FOR KDATA =1 ONLY . . .KDATA =
     1',I18,/)
   26 FORMAT (' INTERMEDIATE OUTPUT EVERY KPRINT(TH) CYCLE  . . KPRINT =
     1',I18,/)
   28 FORMAT (' NUMBER OF KNOWN FIRST MOMENTS . . . . . . . . . . . . . N=
     1',I18,/)
   29 FORMAT (' HIGHER LIMIT  . . . . . . . . . . . . . . . . . . . XMAX =
     1',E18.9,/)
   30 FORMAT (' LOWER LIMIT . . . . . . . . . . . . . . . . . . . . XMIN =
     1',E18.9,/)
   31 FORMAT (' FIRST MOMENTS . . . . . . . . . . . . . . . . . .  CC(I) =
     1',4E18.9,/)
   32 FORMAT (' THE ALLOWED TOLERANCE IN LAGRANGIAN EQUATIONS . . .TOL =
     1',E18.9,/)
   33 FORMAT (' THE CUMULATIVE DISTRIBUTION REQUIRED AT NXP POINTS.NXP =
     1',I18,/)
   34 FORMAT (1H1,//,20X,'INTERMEDIATE RESULTS FOR SUBROUTINE   MEP',/,2
     10X,41('-'),//)
   35 FORMAT (' NUMBER OF INTEGRATION STATION . . . . . . . . . . . .  M =
     1',I18,/)
   36 FORMAT (' MODIFIED MAXIMUM AND MINIMUM LIMITS . .  X2MAX , X2MIN =
     1',2E18.9,/)
   37 FORMAT (' MODIFIED MOMENTS ABOUT THE EXPECTED VALUE . . . .CC(I) =
     1',4E18.9,/)
   38 FORMAT (' MODIFIED MOMENTS ABOUT THE ORIGIN . . . . . . . . . C(I) =
     1',4E18.9,/)
   39 FORMAT (' SUBROUTINE MPOPT TOLERANCES . . . . . . . . . . . ETA(I) =
     1',4E18.9,/)
   40 FORMAT (//,' NORMAL ASSUMPTION STARTING METHOD'/34('-'),/)
   41 FORMAT (' STARTING VALUES   . . . . . . . . . . . . . . . .  AL(I) =
     1',4E18.9,/)
   42 FORMAT (//,' UNIFORM ASSUMPTION STARTING METHOD'/35('-'),/)
   43 FORMAT (//,' N POINTS STARTING METHOD'/25('-'),/)
   44 FORMAT (//,' STEP BY STEP STARTING METHOD'/29('-'),/)
   45 FORMAT (//,'  CYC NUMF   NORMGRAD        TOTAL',24X,'VARIABLES',40
     1X,'RESIDUALS',/,'  NO.',10X,'RESIDUALS      X(1)            X(2)
     2        X(3)            X(4)        R(1)        R(2)        R(3)        R
     3(4)',//)
   46 FORMAT (' THE PROGRAM HAS FAILED')
   47 FORMAT (' THE MODIFIED LAGRANGIAN MULTIPLIERS ARE . . . . . . . . .
     1',4E18.9/57X,4E18.9)
   48 WRITE(6,49)
   49 FORMAT(53H WARNING - MEAN IS NEARLY ZERO AND MEP1 WILL NOT WORK/12
     1H TRANSFORM X)
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUBROUTINE MFOPT (X,NDIM,ETA,EST,MAX,MODE,IPRINT)
C     IMPLICIT REAL*8 (A-H,O-Z)
C     REAL* KTB,IPRINT
      COMMON /FAIL/ NFAIL
      DIMENSION X(*), X1(10), X2(10), G1(10), G2(10), ALFA(10), H(10), P
     1(10,10), Y(10), PY(10), PE(10), ETA(*), BIGV(10), RR(8)
      EXTERNAL FUNCT
      KRST=0
      KTB=0
      IFLAG=0
      M=0
      N2=NDIM+1
      N1=NDIM+2
      NUMF=0
      IER=0
      DO 1 I=1,N1
      X1(I)=X(I)
    1 CONTINUE
      CALL FUNCT (NDIM,X1,F1,G1,RR)
      NUMF=NUMF+1
      DO 2 I=1,NDIM
      X2(I)=X1(I)
      G2(I)=G1(I)
      H(I)=-G1(I)
    2 CONTINUE
      F2=F1
      X2(N2)=X1(N2)
      X2(N1)=X1(N1)
    3 CONTINUE
      KOUNT=0
      EPS=ETA(4)
      CALL LINES (FUNCT,X2,H,RO,NDIM,F2,G2,NUMF,IER,EPS,EST,RR)
      IF (NFAIL.EQ.1) RETURN
      IF (IER.NE.0) GO TO 30
      DO 4 I=1,N1
      BIGV(I)=X2(I)
      ALFA(I)=X2(I)
    4 CONTINUE
      RO=-RO
      GG=0.
      DO 5 I=1,NDIM
      GG=GG+G2(I)*G2(I)
    5 CONTINUE
      GG=SQRT(GG)
      IF (IPRINT.EQ.0) GO TO 7
      IF (MOD(KTB,IPRINT).NE.0) GO TO 6
      CALL OUTP (X2,F2,M,NDIM,GG,NUMF,RR)
    6 KTB=KTB+1
    7 DO 9 I=1,N1
      DO 8 J=1,N1
      P(I,J)=0.
    8 CONTINUE
      P(I,I)=1.
    9 CONTINUE
   10 CONTINUE
      PRINT*,KOUNT
      KOUNT=0
      KOUNT=KOUNT+1
   11 DO 12 I=1,NDIM
      Y(I)=G2(I)
C     PRINT*,'GOT BY A1'
   12 CONTINUE
      Y(N2)=F2
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      Y(N1)=ETA(1)
      V=0.
      DO 13 I=1,NDIM
      V=V+X2(I)*G2(I)
C     PRINT*,'GOT BY A2'
13    CONTINUE
      YA=0.
      DO 14 I=1,N1
      YA=YA+Y(I)*ALFA(I)
C     PRINT*,'GOT BY A3'
14    CONTINUE
      VYA=V-YA
      BIGV(KOUNT)=V
      DO 15 I=1,N1
      PY(I)=0.
      PE(I)=P(I,KOUNT)
      DO 15 J=1,N1
15    PY(I)=PY(I)+P(J,I)*Y(J)
C     PRINT*,'GOT BY A4'
      EPY=PY(KOUNT)
      IF (ABS(EPY).LT.ETA(3)) GO TO 31
      PY(KOUNT)=PY(KOUNT)-1.
      DO 16 I=1,N1
      DO 16 J=1,N1
16    P(I,J)=P(I,J)-PE(I)*PY(J)/EPY
C     PRINT*,'GOT BY A5'
      DO 17 I=1,N1
      ALFA(I)=0.
      DO 17 J=1,N1
17    ALFA(I)=ALFA(I)+P(I,J)*BIGV(J)
C     PRINT*,'GOT BY A6'
      DEL=0.
      DO 18 I=1,NDIM
      DEL=DEL+G2(I)*(X2(I)-ALFA(I))
C     PRINT*,'GOT BY A7'
18    CONTINUE
      IF (ABS(DEL).GT.ETA(4)) GO TO 19
      IF (IFLAG.EQ.1) RETURN
      IFLAG=1
      GO TO 31
19    IFLAG=0
      DO 20 I=1,N1
      H(I)=X2(I)-ALFA(I)
      IF (DEL.GT.0) H(I)=-H(I)
C     PRINT*,'GOT BY A8'
20    CONTINUE
      DO 21 I=1,NDIM
      X1(I)=X2(I)
      G1(I)=G2(I)
C     PRINT*,'GOT BY A9'
21    CONTINUE
      F1=F2
      X1(N2)=X2(N2)
      X1(N1)=X2(N1)
      X2(N2)=ALFA(N2)
      X2(N1)=ALFA(N1)
C     PRINT*,'GOT BY A10'
      CALL LINES (FUNCT,X2,H,RO,NDIM,F2,G2,NUMF,IER,EPS,EST,RR)
C     PRINT*,'GOT BY A11'
      IF (NFAIL.EQ.1) RETURN
C     PRINT*,'GOT BY A12'
      IF (IER.NE.0) GO TO 30
C     PRINT*,'GOT BY A13'
      IF (DEL.GT.0) RO=-RO
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C     PRINT*,'GOT BY A14'
      GG=0.
      DO 22 I=1,NDIM
      GG=GG+G2(I)*G2(I)
C     PRINT*,'GOT BY A15'
22    CONTINUE
      GG=SQRT(GG)
      KOUNT=KOUNT+1
      M=M+1
      IF (IPRINT.EQ.0) GO TO 23
      IF (MOD(KTB,IPRINT).NE.0) GO TO 23
C     PRINT*,'GOT BY G'
      CALL OUTP (X2,F2,M,NDIM,GG,NUMF,RR)
C     PRINT*,'GOT BY H'
23    CONTINUE
      KTB=KTB+1
      IF (MODE.EQ.2) GO TO 25
C     PRINT*,'GOT BY HA'
      IF (M.GT.MAX) GO TO 30
C     PRINT*,'GOT BY HB'
      NSOL=0
      DO 24 I=1,NDIM
      IF (ABS(RR(I)).GT.ETA(2)) NSOL=1
C     PRINT*,'GOT BY HC'
24    CONTINUE
C     PRINT*,'GOT BY HD'
      IF (NSOL.EQ.0) GO TO 26
C     PRINT*,'GOT BY HE'
      GO TO 29
C     PRINT*,'GOT BY HF'
25    IF ((GG.LT.ETA(1)).OR.(M.GT.MAX)) GO TO 26
C     PRINT*,'GOT BY HG'
      GO TO 29
C     PRINT*,'GOT BY HH'
26    CONTINUE
C     PRINT*,'GOT BY HI'
      IF (IPRINT.EQ.0) GO TO 27
C     PRINT*,'GOT BY HJ'
      WRITE (6,33)
C     PRINT*,'GOT BY I'
      CALL OUTP (X2,F2,M,NDIM,GG,NUMF,RR)
C     PRINT*,'GOT BY J'
27    DO 28 I=1,NDIM
      X(I)=X2(I)
28    CONTINUE
      EST=F2
      NFAIL=0
      RETURN
29    CONTINUE
C     PRINT*,KOUNT
C     PRINT*,'GOT BY JA'
      IF (KOUNT.LE.N1) GO TO 11
C     PRINT*,'GOT BY JB'
      GO TO 10
C     PRINT*,'GOT BY JC'
30    PRINT 34, IER
      NFAIL=1
      RETURN
31    KRST=KRST+1
      IF (KRST.GT.10) NFAIL=1
      IF (NFAIL.EQ.1) RETURN
      DO 32 I=1,NDIM
      X1(I)=X2(I)
      G1(I)=G2(I)
```

```
      H(I)=-G1(I)
 32   CONTINUE
      F1=F2
      X1(N2)=X(N2)
      X1(N1)=X(N1)
      X2(N2)=X(N2)
      X2(N1)=X(N1)
      GO TO 3
C
C
 33   FORMAT (/ SOLUTION FOUND')
 34   FORMAT (///,1X,'  THE PROGRAM HAS FAILED---IER = ',I2)
      END

      SUBROUTINE OUTP (XNEW,FQ,KOUNT,N1,GG,NUMF,R)
C     IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION XNEW(*), R(*)
      WRITE (6,6) KOUNT,NUMF,GG,FQ,(XNEW(I),I=1,4),(R(I),I=1,4)
      IF (N1.LT.4) RETURN
      NN=N1-3
      GO TO (1,2,3,4,5), NN
  1   RETURN
  2   WRITE (6,7) XNEW(5),R(5)
      RETURN
  3   WRITE (6,8) (XNEW(I),I=5,6),(R(I),I=5,6)
      RETURN
  4   WRITE (6,9) (XNEW(I),I=5,7),(R(I),I=5,7)
      RETURN
  5   WRITE (6,10) (XNEW(I),I=5,8),(R(I),I=5,8)
      RETURN
C
C
  6   FORMAT (1X,I3,I4,6E14.5,4E10.3)
  7   FORMAT (36X,E14.5,42X,E10.3)
  8   FORMAT (36X,2E14.5,28X,2E10.3)
  9   FORMAT (36X,3E14.5,14X,3E10.3)
 10   FORMAT (36X,4E14.5,4E10.3)
      END

      SUBROUTINE LINES (FUNCT,X,H,AMBDA,N,F,G,NUMF,IER,EPS,EST,RR)
C     IMPLICIT REAL*8 (A-H,O-Z)
C     REAL*8 Z,DX, DY
      COMMON /FAIL/ NFAIL
      DIMENSION H(*), X(*), G(*), RR(*)
      IER=0
      DY=0.
      HNRM=0.
      GNRM=0.
      DO 1 J=1,N
      HNRM=HNRM+ABS(H(J))
      GNRM=GNRM+ABS(G(J))
      DY=DY+H(J)*G(J)
C     PRINT*,'GOT BY B1'
  1   CONTINUE
      IF (DY) 2,31,31
C     PRINT*,'GOT BY B2'
  2   IF (HNRM/GNRM-EPS) 31,31,3
C     PRINT*,'GOT BY B3'
  3   FY=F
      ALFA=2.*(EST-F)/DY
      IF (X(N+1).GT.0.) ALFA=X(N+1)*ALFA/2.
C     PRINT*,'GOT BY B4'
```

```
      AMBDA=1.
      IF (ALFA) 6,6,4
C     PRINT*,'GOT BY B5'
4     IF (ALFA-AMBDA) 5,6,6
C     PRINT*,'GOT BY B6'
5     AMBDA=ALFA
6     ALFA=0.
      DO 8 I=1,N
      X(I)=X(I)+AMBDA*H(I)
C     PRINT*,'GOT BY B7'
8     CONTINUE
7     FX=FY
      DX=DY
C     PRINT*,'GOT BY B8'
      CALL FUNCT (N,X,F,G,RR)
C     PRINT*,'GOT BY B9'
      IF (NFAIL.EQ.1) RETURN
C     PRINT*,'GOT BY B10'
      NUMF=NUMF+1
      IF (F.LT.FX) RETURN
C     PRINT*,'GOT BY B11'
      FY=F
      DY=0.
      DO 9 I=1,N
      DY=DY+G(I)*H(I)
C     PRINT*,'GOT BY B12'
9     CONTINUE
C     PRINT*,'GOT BY B13'
      IF (DY) 10,30,13
C     PRINT*,'GOT BY B14'
10    IF (FY-FX) 11,13,13
C     PRINT*,'GOT BY B15'
11    AMBDA=AMBDA+ALFA
      ALFA=AMBDA
      IF (HNRM*AMBDA-1.E10) 7,7,12
C     PRINT*,'GOT BY B16'
12    IER=2
      GO TO 31
C     PRINT*,'GOT BY B17'
13    T=0.
14    IF (AMBDA) 15,30,15
C     PRINT*,'GOT BY B18'
15    Z=3.*(FX-FY)/AMBDA+DX+DY
      ALFA=AMAX1(ABS(Z),ABS(DX),ABS(DY))
      DALFA=Z/ALFA
      DALFA=DALFA*DALFA-DX/ALFA*DY/ALFA
      IF (DALFA) 31,16,16
C     PRINT*,'GOT BY B19'
16    W=ALFA*SQRT(DALFA)
      ALFA=DY-DX+W+W
      IF (ALFA) 17,18,17
C     PRINT*,'GOT BY B20'
17    ALFA=(DY-Z+W)/ALFA
      GO TO 19
C     PRINT*,'GOT BY B21'
18    ALFA=(Z+DY-W)/(Z+DX+Z+DY)
19    ALFA=ALFA*AMBDA
      DO 20 I=1,N
      X(I)=X(I)+(T-ALFA)*H(I)
20    CONTINUE
      CALL FUNCT (N,X,F,G,RR)
      IF (NFAIL.EQ.1) RETURN
      NUMF=NUMF+1
      IF (F.LT.FX) GO TO 30
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      IF (F-FX) 21,21,22
   21 IF (F-FY) 30,30,22
   22 DALFA=0.
      DO 23 I=1,N
      DALFA=DALFA+G(I)*H(I)
   23 CONTINUE
      IF (DALFA) 24,27,27
   24 IF (F-FX) 26,25,27
   25 IF (DX-DALFA) 26,30,26
   26 FX=F
      DX=DALFA
      T=ALFA
      AMBDA=ALFA
      GO TO 14
   27 IF (FY-F) 29,28,29
   28 IF (DY-DALFA) 29,30,29
   29 FY=F
      DY=DALFA
      AMBDA=AMBDA-ALFA
      GO TO 13
   30 AMBDA=AMBDA-ALFA
      RETURN
   31 CONTINUE
      IF (DY.GE.0.) IER=-2
      IF (GNRM.LE.1.E-10) GO TO 32
      IF (HNRM/GNRM.LE.EPS) IER=-3
   32 CONTINUE
      IF (DALFA.LT.0.) IER=-1
      NFAIL=1
      WRITE(6,33)
   33 FORMAT(///,1X,'  THE PROGRAM HAS FAILED')
      RETURN
C
C
      END
      SUBROUTINE FUNCT (N,AL,U,GRAD,RR)
      IMPLICIT REAL*8 (A-H,O-Z)
C
C
C     THIS  SUBROUTINE IS USED TO CALCULATE THE OPTIMIZATION AND THE
C     GRADIENT AT ANY GIVEN POINT FOR SUBROUTINE POPT
C
      DIMENSION AL(*), GRAD(*), SUM(17), RR(*)
      COMMON /FAIL/ NFAIL
      COMMON/HELP/S(101),XX(16,101),C(8),M
C.... ABOVE LINE CHANGED FROM TEXT

      N21=2*N+1
      ZERO=0.0
      DO 1 I=1,N21
      SUM(I)=0.0
C     PRINT*,'GOT BY C1'
    1 CONTINUE
    2 CONTINUE
      DO 4 I=1,M
      SZ=ZERO
      DO 3 K=1,N
      SZ=SZ+AL(K)*XX(K,I)
C     PRINT*,'GOT BY C2'
    3 CONTINUE
      IF (SZ.GT.74.) GO TO 9
C     PRINT*,'GOT BY C3'
      SS=EXP(SZ)*S(I)
      SUM(1)=SUM(1)+SS
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      DO 4 J=2,N21
      SUM(J)=SUM(J)+XX(J-1,I)*SS
C     PRINT*,'GOT BY C4'
    4 CONTINUE
      DO 5 I=2,N21
      SUM(I)=SUM(I)/SUM(1)
C     PRINT*,'GOT BY C5'
    5 CONTINUE
      U=0.0
      DO 6 I=1,N
      RR(I)=(SUM(I+1)-C(I))/C(I)
      U=U+RR(I)*RR(I)
C     PRINT*,'GOT BY C6'
    6 CONTINUE
      DO 8 K=1,N
      GRAD(K)=0.0
      DO 7 J=1,N
      GRAD(K)=GRAD(K)+(SUM(J+K+1)-SUM(J+1)*SUM(K+1))*RR(J)/C(J)
C     PRINT*,'GOT BY G7'
    7 CONTINUE
      GRAD(K)=GRAD(K)*2.
C     PRINT*,'GOT BY C8'
    8 CONTINUE
C     PRINT*,'GOT BY C9'
      RETURN
C     PRINT*,'GOT BY C10'
    9 CONTINUE
      AA=SZ-32.
      ZERO=ZERO-AA
      GO TO 2
C
C     PRINT*,'GOT BY C11'
      END
      SUBROUTINE START (XMAX,XMIN,ALAMDA,KSTART,CC,NL,IPRINT,UMIN,MODE,M
     1AXFN,ETA)
C     IMPLICIT REAL*8 (A-H,O-Z)
C
C     THIS SUBROUTINE IS USED TO FIND A REASONABLE STARTING POINT FOR
C     SUBROUTINE  MPOPT
C
      DIMENSION R(11)
      DIMENSION CC(*), ETA(*)
      DIMENSION ALAMDA(*), X(10), Y(10), W(10,10)
      COMMON/HELP/S(101),XX(16,101),C(8),M
C.... ABOVE LINE CHANGED FROM TEXT
      COMMON /FAIL/ NFAIL
      GO TO (3,1,5,26), KSTART
    1 CONTINUE
      NFAIL=0
      DO 2 I=1,NL
      ALAMDA(I)=0.0
    2 CONTINUE
      RETURN
    3 CONTINUE
      NFAIL=0
      ALAMDA(1)=CC(1)/CC(2)
      ALAMDA(2)=-.5/CC(2)
      DO 4 I=3,NL
      ALAMDA(I)=0.0
    4 CONTINUE
      RETURN
    5 CONTINUE
```

```
      NFAIL=0
      NNN=NL/2
      NNN=NNN*2
      NP1=NL+1
      DELTA=(XMAX-XMIN)/FLOAT(NL)
      DO 6 I=1,NP1
      X(I)=XMIN+FLOAT(I-1)*DELTA
    6 CONTINUE
      IF (NNN.NE.NL) GO TO 19
      W(1,1)=1.
      W(1,NP1)=1.
      DO 7 I=2,NL,2
      W(1,I)=4.
    7 CONTINUE
      IF (NL.EQ.2) GO TO 9
      NM1=NL-1
      DO 8 I=3,NM1,2
      W(1,I)=2.
    8 CONTINUE
    9 CONTINUE
      DO 10 J=1,NP1
      DO 10 I=2,NP1
   10 W(I,J)=W(I-1,J)*X(J)
      Y(1)=3./DELTA
      DO 11 I=1,NL
      Y(I+1)=C(I)*Y(1)
   11 CONTINUE
      CALL SOLVE (W,Y,XID,NP1,10)
   12 CONTINUE
      DO 13 I=1,NP1
      DO 13 J=1,NP1
   13 W(I,J)=.0
      DO 14 I=1,NP1
      IF (Y(I).LE.0.0) Y(I)=.0002
   14 CONTINUE
      DO 15 I=1,NP1
      Y(I)=ALOG(Y(I))
   15 CONTINUE
      DO 16 I=1,NP1
      W(I,1)=1.
   16 CONTINUE
      DO 17 I=2,NP1
      DO 17 J=1,NP1
   17 W(J,I)=W(J,I-1)*X(J)
      CALL SOLVE (W,Y,XID,NP1,10)
      DO 18 I=1,NL
      ALAMDA(I)=Y(I+1)
   18 CONTINUE
      RETURN
   19 CONTINUE
      R(1)=3./8.
      R(4)=3./8.
      R(2)=9./8.
      R(3)=9./8.
      IF (NL.EQ.3) GO TO 22
      R(NL+1)=1./3.
      R(4)=R(4)+1./3.
      DO 20 I=5,NL,2
      R(I)=4./3.
   20 CONTINUE
      IF (NL.EQ.5) GO TO 22
      NS=NL-1
      DO 21 I=6,NS,2
      R(I)=2./3.
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
21    CONTINUE
22    CONTINUE
      DO 23 I=1,NP1
      W(1,I)=R(I)
23    CONTINUE
      DO 24 J=1,NP1
      DO 24 I=2,NP1
24    W(I,J)=W(I-1,J)*X(J)
      Y(1)=1./DELTA
      DO 25 I=1,NL
      Y(I+1)=C(I)*Y(1)
25    CONTINUE
      CALL SOLVE (W,Y,XID,NP1,10)
      GO TO 12
26    CONTINUE
      N=2
      ALAMDA(2)=-.5/CC(2)
      ALAMDA(1)=CC(1)/CC(2)
      NFAIL=0
27    CONTINUE
      ALAMDA(N+1)=2.0
      ALAMDA(N+2)=0.0
C     PRINT*,'GOT BY A'
      CALL MPOPT (ALAMDA,N,ETA,UMIN,MAXFN,MODE,IPRINT)
C     PRINT*,'GOT BY B'
      IF (NFAIL.EQ.1) RETURN
      IF (N.EQ.NL) RETURN
      ALAMDA(N+1)=0.0
      N=N+1
      GO TO 27
      END

      SUBROUTINE SOLVE (A,X,XID,N,NA)
C     IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION A(NA,*), X(*)
      D=0.
      DATA DIV/.693147181/
      DO 6 I=1,N
      AA=0.
      DO 1 J=I,N
      AB=ABS(A(J,I))
      IF (AB.LE.AA) GO TO 1
      K=J
      AA=AB
1     CONTINUE
      D=D+ALOG(AA)
      IF (I.EQ.N) GO TO 7
      IF (K.EQ.I) GO TO 3
      DO 2 J=I,N
      AB=A(I,J)
      A(I,J)=A(K,J)
      A(K,J)=AB
2     CONTINUE
      AB=X(I)
      X(I)=X(K)
      X(K)=AB
3     I1=I+1
      DO 5 J=I1,N
      AA=-A(J,I)/A(I,I)
      A(J,I)=0.
      DO 4 K=I1,N
      A(J,K)=A(J,K)+AA*A(I,K)
4     CONTINUE
```

ORIGINAL PAGE IS OF POOR QUALITY

```
      X(J)=X(J)+AA*X(I)
 5    CONTINUE
 6    CONTINUE
 7    XID=D/DIV
      X(N)=X(N)/A(N,N)
      DO 9 II=2,N
      I=N+1-II
      I1=I+1
      AA=0.
      DO 8 J=I1,N
      AA=AA+A(I,J)*X(J)
 8    CONTINUE
      X(I)=(X(I)-AA)/A(I,I)
 9    CONTINUE
      RETURN
      END

      SUBROUTINE SIMSON
C     IMPLICIT REAL*8 (A-H,O-Z)
C
C     THIS SUBROUTINE IS TO CALCULATE THE SIMPSON MULTIPLIERS
C
      COMMON/HELP/S(101),XX(16,101),C(8),M
C.... ABOVE LINE CHANGED FROM TEXT
      S(1)=1.
      S(M)=1.
      N=M-1
      DO 1 I=2,N,2
      S(I)=4.
 1    CONTINUE
      N=N-1
      DO 2 I=3,N,2
      S(I)=2.
 2    CONTINUE
      RETURN
      END

      SUBROUTINE MULTI (XMAX,XMIN,N)
C     IMPLICIT REAL*8 (A-H,O-Z)
C
C     THIS SUBROUTINE IS USED TO GENERATE THE X,S POWER FOR SUBROUTINE
C     FUNCT
C
      COMMON/HELP/S(101),XX(16,101),C(8),M
C.... ABOVE LINE CHANGED FROM TEXT
      DELTA=(XMAX-XMIN)/FLOAT(M-1)
      DO 1 I=1,M
      XX(1,I)=XMIN+FLOAT(I-1)*DELTA
      NN=2*N
      DO 1 J=2,NN
      XX(J,I)=XX(J-1,I)*XX(1,I)
 1    CONTINUE
      RETURN
      END

      SUBROUTINE CONVER (CM,NL)
C     IMPLICIT REAL*8 (A-H,O-Z)
C
C     THIS SUBROUTINE IS TO CALCULATE THE MOMENTS ABOUT THE ORIGIN
C
      DIMENSION CM(*)
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
      COMMON/HELP/S(101),XX(16,101),C(8),M
C.... ABOVE LINE CHANGED FROM TEXT
      C(1)=CM(1)
      IF (NL.EQ.1) RETURN
      DO 2 J=2,NL
      C(J)=CM(J)-C(1)**J*(-1.)**J
      N=J-1
      DO 1 K=1,N
      C(J)=C(J)-(-1.)**K*FACTO(J)/(FACTO(K)*FACTO(J-K))*C(1)**(K)*C(J-K)
1     CONTINUE
2     CONTINUE
      RETURN
      END

      SUBROUTINE TRN1 (X1MAX,X1MIN,C,X2MAX,X2MIN,NL)
      IMPLICIT REAL*8 (A-H,O-Z)
C
C     THIS SUBROUTINE IS USED TO CALCULATE THE MOMENTS FOR THE MODIFIED
C     LIMITS
C
      DIMENSION C(1)
      SCL=(X1MAX-X1MIN)/(X2MAX-X2MIN)
      C(1)=C(1)/SCL-X1MIN/SCL+X2MIN
      IF (NL.EQ.1) RETURN
      DO 1 I=2,NL
      C(I)=C(I)/SCL**(FLOAT(I))
1     CONTINUE
      RETURN
      END

C      SUBROUTINE TRN2(X1MAX,X1MIN,X,X2MAX,X2MIN,N)
C      IMPLICIT REAL*8 (A-H,O-Z)
C.... THIS SUBROUTINE IS AN ALTERNATIVE TO TRN2 (BELOW)
C....
C.... CALCULATES THE LAGRANGIAN MULTIPLIERS FOR A DIFFERENT INTERVAL
C.... DOUBLE PRECISION VERSION
C      DOUBLE PRECISION S,A,DX(10),FAC,DX1MAX,DX1MIN,DX2MAX,DX2MIN
C      DIMENSION X(*)
C      DX1MAX=X1MAX
C      DX1MIN=X1MIN
C      DX2MAX=X2MAX
C      DX2MIN=X2MIN
C      NP1=N+1
C      DO 10 I=1,NP1
C10    DX(I)=X(I)
C      S=(DX1MAX-DX1MIN)/(DX2MAX-DX2MIN)
C      A=DX2MIN-DX1MIN/S
C      DX(1)=DX(1)-ALOG(S)
C      DO 1 I=1,N
C      DX(1)=DX(1)+DX(I+1)*A**I
C1     CONTINUE
C      IF(N.EQ.1)GO TO 6
C      DO 5 J=2,N
C      DO 3 I=J,N
C      FAC=1.
C      KK=I-J+2
C      DO 2 K=KK,I
C      FAC=FAC*DBLE(FLOAT(K))
C2     CONTINUE
C      DX(J)=DX(J)+FAC/DBLE(FACTO(J-1))*A**(I-J+1)*DX(I+1)
C3     CONTINUE
C4     DX(J)=DX(J)/S**(J-1)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C5      CONTINUE
C6      CONTINUE
C       DX(N+1)=DX(N+1)/S**N
C       DO 11 I=1,NP1
C11     X(I)=DX(I)
C       RETURN
C       END

        SUBROUTINE TRN2 (X1MAX,X1MIN,X,X2MAX,X2MIN,N)
C       IMPLICIT REAL*8 (A-H,O-Z)
C
C       THIS SUBROUTINE IS USED TO CALCULATE THE LAGRANGIAN MULTIPLIERS
C       AT  THE ORIGINAL LIMITS
C
        DIMENSION X(1)
        S=(X1MAX-X1MIN)/(X2MAX-X2MIN)
        A=X2MIN-X1MIN/S
        X(1)=X(1)-ALOG(S)
        DO 1 I=1,N
        X(1)=X(1)+X(I+1)*A**I
  1     CONTINUE
        IF (N.EQ.1) GO TO 6
        DO 5 J=2,N
        DO 3 I=J,N
        FAC=1.
        KK=I-J+2
        DO 2 K=KK,I
        FAC=FAC*FLOAT(K)
  2     CONTINUE
        X(J)=X(J)+FAC/FACTO(J-1)*A**(I-J+1)*X(I+1)
  3     CONTINUE
  4     X(J)=X(J)/S**(J-1)
  5     CONTINUE
  6     CONTINUE
        X(N+1)=X(N+1)/S**N
        RETURN
        END

        FUNCTION CDF (XMIN,XMAX,XP,AL,N)
C       IMPLICIT REAL*8 (A-H,O-Z)
C       THIS FUNCTION SUBROUTINE IS TO CALCULATE THE CUMMULATIVE DISTRIBU-
C       TION FUNCTION AT A GIVEN POINT
C....   INPUT
C         XMIN = LOWER BOUND
C         XMAX = UPPER BOUND
C         XP = SPECIFIED POINT
C         AL(I) = ARRAY OF PARAMETERS, DIMENSION N
C         N = NUMBER OF PARAMETERS
        DIMENSION AL(*)
        IF (XP.LE.XMIN) GO TO 3
        IF (XP.GE.XMAX) GO TO 4
        RANGE=XMAX-XMIN
        RANGEN=XP-XMIN
        SS=RANGEN/RANGE*51.
        JSS=SS
        JSS=(JSS/2)*2+5
        AREA=0.0
        JSM1=JSS-1
        DELTA=RANGEN/FLOAT(JSM1)
        DO 1 I=2,JSM1,2
        X=XMIN+FLOAT(I-1)*DELTA
        AREA=AREA+4.*ENTRPF(AL,N,X)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
1     CONTINUE
      JSM1=JSM1-1
      DO 2 I=3,JSM1,2
      X=XMIN+FLOAT(I-1)*DELTA
      AREA=AREA+2.*ENTRPF(AL,N,X)
2     CONTINUE
      AREA=AREA+ENTRPF(AL,N,XMIN)+ENTRPF(AL,N,XP)
      AREA=AREA*DELTA/3.
      CDF=AREA
      GO TO 5
3     CDF=0.0
      GO TO 5
4     CDF=1.
5     CONTINUE
      RETURN
      END

      FUNCTION ENTRPF (AL,NPL,X)
C     IMPLICIT REAL*8 (A-H,O-Z)
C
C     FUNCTION TO EVALUATE THE ENTROPY DENSITY FUNCTION AT A GIVEN POINT
C.... INPUT
C       AL(I) = ARRAY CONTAINING PARAMETERS, DIMENSION NPL
C       NPL = NUMBER OF PARAMETERS
C       X = GIVEN VALUE
C
      DIMENSION AL(*)
      S=AL(1)
      DO 1 I=2,NPL
      S=S+AL(I)*X**(I-1)
      CONTINUE
      ENTRPF=EXP(S)
      RETURN
      END

      FUNCTION FACTO (M)
C     IMPLICIT REAL*8 (A-H,O-Z)

C.... CALCULATES FACTORIAL OF M
      FACTO=1.
      IF (M.EQ.0) RETURN
      DO 1 I=1,M
      FACTO=FACTO*FLOAT(I)
1     CONTINUE
      RETURN
      END
```

```
EEEEEEEEEE    44444444444444444444444444444444444444444444444444444444444444444444444444444444444444444444444    EEEEEEEE
EEEEEEEEEE    444444444444444444444444  Digital Equipment Corporation - VAX/VMS Version V4.7  4444444444444444444444444    EEEEEEEE
EEEEEEEEEE    44444444444444444444444444444444444444444444444444444444444444444444444444444444444444444444444    EEEEEEEE
```

DECNET

NORMAL

INP;39

ORIGINAL PAGE IS
OF POOR QUALITY

111

```
File _DUA0:[]NORMAL.INP;39 (445,121,0), last revised on 22-DEC-1988 13:25, is a 1 block sequential file owned by UIC [DECNET].  The
records are variable length with implied (CR) carriage control.  The longest record is 24 bytes.

Job NORMAL (132) queued to TERM$LA120A on 22-DEC-1988 13:26 by user DECNET, UIC [DECNET], under account DECNET at priority 100,
started on printer LTA4: on 22-DEC-1988 13:30 from queue TERM$LA120A.

EEEEEEEEEE    44444444444444444444444444444444444444444444444444444444444444444444444444444444444444444444444    EEEEEEEE
EEEEEEEEEE    444444444444444444444444  Digital Equipment Corporation - VAX/VMS Version V4.7  4444444444444444444444444    EEEEEEEE
EEEEEEEEEE    44444444444444444444444444444444444444444444444444444444444444444444444444444444444444444444444    EEEEEEEE
```

ORIGINAL PAGE IS
OF POOR QUALITY

1	40
900.0000	45.0000
8.0000	0.8000
500.0000	25.0000
7.0000	0.7000
250.0000	12.5000
20.0000	1.0000
150.0000	7.5000
0.5000	0.0150
1500.0000	75.0000
20.0000	0.6000
850.0000	25.5000

```
ZZZZZZZZZZ   4444444444444444444444444444444444444444444444444444444444444444444444444444444444444444   ZZZZZZZZZZ
ZZZZZZZZZZ   44444444444444444444444  Digital Equipment Corporation - VAX/VMS Version V4.7  4444444444444444444444   ZZZZZZZZZZ
ZZZZZZZZZZ   4444444444444444444444444444444444444444444444444444444444444444444444444444444444444444   ZZZZZZZZZZ
```

NETNONPRIV

RANDM44

CPR;1

File _DBA0:[]RANDM44.CPR;1 (383,942,0), last revised on 23-NOV-1988 11:21, is a 51 block sequential file owned by UIC [11,11]. The records are variable length with FORTRAN (FTN) carriage control. The longest record is 132 bytes.

Job RANDM44 (685) queued to SYS$SBPRT on 23-NOV-1988 11:22 by user NETNONPRIV, UIC [11,11], under account 2010OADD at priority 100, started on printer _TTF7: on 23-NOV-1988 11:22 from queue TTF7.

```
ZZZZZZZZZZ   4444444444444444444444444444444444444444444444444444444444444444444444444444444444444444   ZZZZZZZZZZ
ZZZZZZZZZZ   44444444444444444444444  Digital Equipment Corporation - VAX/VMS Version V4.7  4444444444444444444444   ZZZZZZZZZZ
ZZZZZZZZZZ   4444444444444444444444444444444444444444444444444444444444444444444444444444444444444444   ZZZZZZZZZZ
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
1  40
 0.9000E+03  0.4500E+02
OGNORMAL  SF
  0.8165E+03  0.9306E+03  0.8813E+03  0.8726E+03  0.9717E+03
  0.8942E+03  0.8963E+03  0.8590E+03  0.8760E+03  0.8505E+03
  0.9148E+03  0.9603E+03  0.9484E+03  0.8751E+03  0.9161E+03
  0.8845E+03  0.8782E+03  0.8993E+03  0.9872E+03  0.8657E+03
  0.1010E+04  0.9801E+03  0.9605E+03  0.8738E+03  0.9339E+03
  0.8546E+03  0.8574E+03  0.9446E+03  0.9070E+03  0.9341E+03
  0.8816E+03  0.9039E+03  0.8584E+03  0.8706E+03  0.8554E+03
  0.9232E+03  0.9712E+03  0.8108E+03  0.9656E+03  0.9446E+03
1  40
 0.8000E+01  0.8000E+00
LOGNORMAL  XLNMF
  0.6570E+01  0.8531E+01  0.7652E+01  0.7503E+01  0.9299E+01
  0.7878E+01  0.7916E+01  0.7271E+01  0.7560E+01  0.7128E+01
  0.8245E+01  0.9083E+01  0.8859E+01  0.7546E+01  0.8268E+01
  0.7708E+01  0.7599E+01  0.7967E+01  0.9599E+01  0.7384E+01
  0.1005E+02  0.7632E+01  0.9088E+01  0.7524E+01  0.8591E+01
  0.7197E+01  0.7243E+01  0.8789E+01  0.8104E+01  0.8596E+01
  0.7658E+01  0.8049E+01  0.7261E+01  0.7469E+01  0.7209E+01
  0.8397E+01  0.9290E+01  0.6608E+01  0.9184E+01  0.8790E+01
1  40
 0.5000E+03  0.2500E+02
LOGNORMAL  SO
  0.4536E+03  0.5170E+03  0.4896E+03  0.4848E+03  0.5398E+03
  0.4968E+03  0.4980E+03  0.4772E+03  0.4866E+03  0.4725E+03
  0.5082E+03  0.5335E+03  0.5269E+03  0.4862E+03  0.5090E+03
  0.4914E+03  0.4879E+03  0.4996E+03  0.5485E+03  0.4809E+03
  0.5613E+03  0.4890E+03  0.5336E+03  0.4855E+03  0.5188E+03
  0.4748E+03  0.4763E+03  0.5248E+03  0.5039E+03  0.5190E+03
  0.4898E+03  0.5022E+03  0.4769E+03  0.4837E+03  0.4752E+03
  0.5129E+03  0.5396E+03  0.4549E+03  0.5365E+03  0.5248E+03
1  40
 0.7000E+01  0.7000E+00
LOGNORMAL  XLNMO
  0.5749E+01  0.7465E+01  0.6696E+01  0.6565E+01  0.8136E+01
  0.6894E+01  0.6926E+01  0.6362E+01  0.6615E+01  0.6237E+01
  0.7214E+01  0.7947E+01  0.7752E+01  0.6602E+01  0.7235E+01
  0.6744E+01  0.6649E+01  0.6971E+01  0.8399E+01  0.6461E+01
  0.8797E+01  0.6678E+01  0.7952E+01  0.6583E+01  0.7517E+01
  0.6297E+01  0.6338E+01  0.7690E+01  0.7091E+01  0.7521E+01
  0.6701E+01  0.7043E+01  0.6353E+01  0.6535E+01  0.6308E+01
  0.7347E+01  0.8129E+01  0.5782E+01  0.8036E+01  0.7691E+01
1  40
 0.2500E+03  0.1250E+02
LOGNORMAL  S
  0.2268E+03  0.2585E+03  0.2448E+03  0.2424E+03  0.2699E+03
  0.2484E+03  0.2490E+03  0.2386E+03  0.2433E+03  0.2363E+03
  0.2541E+03  0.2667E+03  0.2634E+03  0.2431E+03  0.2545E+03
  0.2457E+03  0.2439E+03  0.2498E+03  0.2742E+03  0.2405E+03
  0.2807E+03  0.2445E+03  0.2668E+03  0.2427E+03  0.2594E+03
  0.2374E+03  0.2382E+03  0.2624E+03  0.2519E+03  0.2595E+03
  0.2449E+03  0.2511E+03  0.2385E+03  0.2418E+03  0.2376E+03
  0.2565E+03  0.2698E+03  0.2275E+03  0.2682E+03  0.2624E+03
1  40
 0.2000E+02  0.1000E+01
LOGNORMAL  SIGO
-0.1814E+02 -0.2068E+02 -0.1958E+02 -0.1939E+02 -0.2159E+02
-0.1987E+02 -0.1992E+02 -0.1909E+02 -0.1947E+02 -0.1890E+02
```

```
-0.2033E+02 -0.2134E+02 -0.2107E+02 -0.1945E+02 -0.2036E+02
-0.1965E+02 -0.1952E+02 -0.1998E+02 -0.2194E+02 -0.1924E+02
-0.2245E+02 -0.1956E+02 -0.2135E+02 -0.1942E+02 -0.2075E+02
-0.1899E+02 -0.1905E+02 -0.2099E+02 -0.2015E+02 -0.2076E+02
-0.1959E+02 -0.2009E+02 -0.1908E+02 -0.1935E+02 -0.1901E+02
-0.2052E+02 -0.2158E+02 -0.1820E+02 -0.2146E+02 -0.2099E+02
1  40
 0.1500E+03  0.7500E+01
LOGNORMAL SIG
 0.1361E+03  0.1551E+03  0.1469E+03  0.1454E+03  0.1619E+03
 0.1490E+03  0.1494E+03  0.1432E+03  0.1460E+03  0.1418E+03
 0.1525E+03  0.1600E+03  0.1581E+03  0.1459E+03  0.1527E+03
 0.1474E+03  0.1464E+03  0.1499E+03  0.1645E+03  0.1443E+03
 0.1684E+03  0.1467E+03  0.1601E+03  0.1456E+03  0.1556E+03
 0.1424E+03  0.1429E+03  0.1574E+03  0.1512E+03  0.1557E+03
 0.1469E+03  0.1507E+03  0.1431E+03  0.1451E+03  0.1426E+03
 0.1539E+03  0.1619E+03  0.1365E+03  0.1609E+03  0.1574E+03
1  40
 0.5000E+00  0.1500E-01
NORMAL XXN
 0.4352E+00  0.4832E+00  0.5104E+00  0.4984E+00  0.5012E+00
 0.4884E+00  0.4749E+00  0.5070E+00  0.5070E+00  0.5227E+00
 0.4956E+00  0.5007E+00  0.5144E+00  0.4727E+00  0.4758E+00
 0.5011E+00  0.5066E+00  0.4637E+00  0.4956E+00  0.4775E+00
 0.4969E+00  0.5073E+00  0.5034E+00  0.5222E+00  0.5153E+00
 0.5010E+00  0.4801E+00  0.5059E+00  0.4968E+00  0.5079E+00
 0.5201E+00  0.5107E+00  0.4905E+00  0.4750E+00  0.5095E+00
 0.4933E+00  0.5051E+00  0.5104E+00  0.5355E+00  0.4948E+00
1  40
NORMAL XXM
 0.4352E+00  0.4832E+00  0.5104E+00  0.4984E+00  0.5012E+00
 0.4884E+00  0.4749E+00  0.5070E+00  0.5070E+00  0.5227E+00
 0.4956E+00  0.5007E+00  0.5144E+00  0.4727E+00  0.4758E+00
 0.5011E+00  0.5066E+00  0.4637E+00  0.4956E+00  0.4775E+00
 0.4969E+00  0.5073E+00  0.5034E+00  0.5222E+00  0.5153E+00
 0.5010E+00  0.4801E+00  0.5059E+00  0.4968E+00  0.5079E+00
 0.5201E+00  0.5107E+00  0.4905E+00  0.4750E+00  0.5095E+00
 0.4933E+00  0.5051E+00  0.5104E+00  0.5355E+00  0.4948E+00
1  40
NORMAL XXG
 0.4352E+00  0.4832E+00  0.5104E+00  0.4984E+00  0.5012E+00
 0.4884E+00  0.4749E+00  0.5070E+00  0.5070E+00  0.5227E+00
 0.4956E+00  0.5007E+00  0.5144E+00  0.4727E+00  0.4758E+00
 0.5011E+00  0.5066E+00  0.4637E+00  0.4956E+00  0.4775E+00
 0.4969E+00  0.5073E+00  0.5034E+00  0.5222E+00  0.5153E+00
 0.5010E+00  0.4801E+00  0.5059E+00  0.4968E+00  0.5079E+00
 0.5201E+00  0.5107E+00  0.4905E+00  0.4750E+00  0.5095E+00
 0.4933E+00  0.5051E+00  0.5104E+00  0.5355E+00  0.4948E+00
1  40
 0.1500E+04  0.7500E+02
NORMAL TF
 0.1176E+04  0.1416E+04  0.1552E+04  0.1492E+04  0.1506E+04
 0.1442E+04  0.1374E+04  0.1535E+04  0.1535E+04  0.1613E+04
 0.1478E+04  0.1504E+04  0.1572E+04  0.1364E+04  0.1379E+04
 0.1506E+04  0.1533E+04  0.1318E+04  0.1478E+04  0.1388E+04
 0.1484E+04  0.1537E+04  0.1517E+04  0.1611E+04  0.1577E+04
 0.1505E+04  0.1400E+04  0.1530E+04  0.1484E+04  0.1540E+04
 0.1601E+04  0.1554E+04  0.1452E+04  0.1375E+04  0.1547E+04
 0.1467E+04  0.1525E+04  0.1552E+04  0.1678E+04  0.1474E+04
1  40
```

ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

```
  0.2000E+02   0.6000E+00
 NORMAL  TO
  0.1741E+02   0.1933E+02   0.2042E+02   0.1994E+02   0.2005E+02
  0.1953E+02   0.1900E+02   0.2028E+02   0.2028E+02   0.2091E+02
  0.1982E+02   0.2003E+02   0.2057E+02   0.1891E+02   0.1903E+02
  0.2004E+02   0.2027E+02   0.1855E+02   0.1982E+02   0.1910E+02
  0.1988E+02   0.2029E+02   0.2013E+02   0.2089E+02   0.2061E+02
  0.2004E+02   0.1920E+02   0.2024E+02   0.1987E+02   0.2032E+02
  0.2081E+02   0.2043E+02   0.1962E+02   0.1900E+02   0.2038E+02
  0.1973E+02   0.2020E+02   0.2042E+02   0.2142E+02   0.1979E+02
1   40
  0.8500E+03   0.2550E+02
 NORMAL  T
  0.7399E+03   0.8215E+03   0.8677E+03   0.8474E+03   0.8521E+03
  0.8302E+03   0.8073E+03   0.8618E+03   0.8619E+03   0.8885E+03
  0.8424E+03   0.8512E+03   0.8744E+03   0.8037E+03   0.8089E+03
  0.8519E+03   0.8613E+03   0.7882E+03   0.8424E+03   0.8118E+03
  0.8447E+03   0.8624E+03   0.8557E+03   0.8877E+03   0.9760E+03
  0.8517E+03   0.8161E+03   0.8601E+03   0.8446E+03   0.8635E+03
  0.8842E+03   0.8682E+03   0.8338E+03   0.8074E+03   0.8661E+03
  0.8387E+03   0.8586E+03   0.8677E+03   0.9104E+03   0.8412E+03
 LOG OF CYCLES TO REACH MEAN FATIGUE STR =
 50 MPA
  0.6026E+01   0.7799E+01   0.6977E+01   0.6849E+01   0.8486E+01
  0.7199E+01   0.7242E+01   0.6632E+01   0.6896E+01   0.6491E+01
  0.7529E+01   0.8289E+01   0.8074E+01   0.6904E+01   0.7564E+01
  0.7034E+01   0.6931E+01   0.7295E+01   0.8765E+01   0.6754E+01
  0.9179E+01   0.6961E+01   0.8292E+01   0.6851E+01   0.7828E+01
  0.6568E+01   0.6623E+01   0.8017E+01   0.7399E+01   0.7839E+01
  0.6975E+01   0.7339E+01   0.6634E+01   0.6833E+01   0.6574E+01
  0.7669E+01   0.8475E+01   0.6025E+01   0.8351E+01   0.8027E+01
 SORTED LOG OF CYCLES
  0.6025E+01   0.6026E+01   0.6491E+01   0.6568E+01   0.6574E+01
  0.6623E+01   0.6632E+01   0.6634E+01   0.6754E+01   0.6833E+01
  0.6849E+01   0.6851E+01   0.6896E+01   0.6904E+01   0.6931E+01
  0.6961E+01   0.6975E+01   0.6977E+01   0.7034E+01   0.7199E+01
  0.7242E+01   0.7295E+01   0.7339E+01   0.7399E+01   0.7529E+01
  0.7564E+01   0.7669E+01   0.7799E+01   0.7828E+01   0.7839E+01
  0.8017E+01   0.8027E+01   0.8074E+01   0.8289E+01   0.8292E+01
  0.8351E+01   0.8475E+01   0.8486E+01   0.8765E+01   0.9179E+01
 SAMPLE MOMENTS
  0.7355E+01   0.5703E+00   0.1782E+00   0.7624E+00
 BNDS(1),BNDS(2)=   0.5723E+01    0.9638E+01
```

```
                    INPUT DATA FOR SUBROUTINE   MEP1

INPUT DATA IS PRINTED OUT FOR KDATA =1 ONLY . . .KDATA =              1
INTERMEDIATE OUTPUT EVERY KPRINT(TH) CYCLE  . . KPRINT =              1
NUMBER OF KNOWN FIRST MOMENTS . . .  . . . . . . . . N=               4
HIGHER LIMIT . . . . . . . . . . . . . . . . . XMAX =  0.963779301E+01
LOWER LIMIT . . . . . . . . . . . . . . . . . . XMIN =  0.572349819E+01
FIRST MOMENTS . . . . . . . . . . . . . . . . . CC(I) =  0.735481628E+01   0.570334345E+00   0.178168965E+00   0.762378196E+00
THE ALLOWED TOLERANCE IN LAGRANGIAN EQUATIONS . . . TOL =  0.100000000E-05
THE CUMULATIVE DISTRIBUTION REQUIRED AT NXP POINTS NXP =              0
```

ORIGINAL PAGE IS
OF POOR QUALITY

INTERMEDIATE RESULTS FOR SUBROUTINE MEP

NUMBER OF INTEGRATION STATION	M =	31				
MODIFIED MAXIMUM AND MINIMUM LIMITS	X2MAX , X2MIN =	0.100000000E+01	0.000000000E+00			
MODIFIED MOMENTS ABOUT THE EXPECTED VALUE	CC(I) =	0.416759124E+00	0.372239503E-01	0.297078661E-02	0.324755365E-02	
MODIFIED MOMENTS ABOUT THE ORIGIN	C(I) =	0.416759124E+00	0.210912118E+00	0.121897178E+00	0.771597013E-01	
SUBROUTINE MPOPT TOLERANCES	ETA(I) =	0.100000000E-11	0.100000000E-05	0.100000000E-23	0.100000000E-23	

NORMAL ASSUMPTION STARTING METHOD

STARTING VALUES	AL(I) =	0.111959940E+02	-0.134322122E+02	0.000000000E+00	0.000000000E+00

CYC NO.	NUMF	NORMGRAD RESIDUALS	TOTAL X(1)	VARIABLES X(2)	X(3)	X(4)	R(1)	RESIDUALS R(2)	R(3)	R(4)	
0	2	0.17606E-01	0.23715E-02	0.11222E+02	-0.13405E+02	0.23839E-01	0.19798E-01	0.224E-01	0.210E-01	-0.417E-02	-0.376E-02
1	4	0.32974E-02	0.22064E-02	0.11262E+02	-0.13405E+02	0.23839E-01	0.19798E-01	0.256E-01	0.265E-01	0.309E-02	-0.290E-01
2	5	0.51891E-02	0.97080E-04	0.80751E+01	-0.10197E+02	0.23839E-01	0.19798E-01	-0.809E-02	0.106E-02	0.[illegible]E-02	0.441E-02
3	7	0.19418E-02	0.92570E-04	0.77949E+01	-0.95481E+01	-0.41358E+00	0.19798E-01	-0.894E-02	0.213E-03	0.225E-02	0.273E-02
4	8	0.42977E-02	0.31381E-04	0.21444E+02	-0.62921E+02	0.79107E+02	-0.39801E+02	-0.382E-02	0.173E-02	0.2[illegible]E-02	0.231E-02
5	10	0.27787E-02	0.23060E-04	0.17639E+02	-0.47370E+02	0.55444E+02	-0.27847E+02	-0.283E-02	0.294E-02	0.25[illegible]E-02	0.563E-04
6	11	0.24290E-02	0.18863E-04	0.20061E+02	-0.56621E+02	0.69056E+02	-0.34621E+02	-0.200E-02	0.304E-02	0.232E-02	-0.444E-03
7	13	0.10508E-02	0.10587E-04	0.22892E+02	-0.67284E+02	0.84353E+02	-0.42016E+02	-0.254E-02	0.126E-02	0.569E-03	-0.190E-02
8	15	0.13076E-02	0.10539E-04	0.23238E+02	-0.68597E+02	0.86281E+02	-0.42975E+02	-0.248E-02	0.120E-02	0.493E-03	-0.167E-02
9	16	0.35516E-03	0.36451E-05	0.26844E+02	-0.79546E+02	0.98742E+02	-0.47577E+02	-0.164E-02	0.351E-04	-0.180E-03	0.966E-03
10	19	0.12140E-03	0.36089E-05	0.27017E+02	-0.80287E+02	0.99926E+02	-0.48205E+02	-0.169E-02	-0.380E-04	-0.258E-03	0.831E-03
11	20	0.27919E-03	0.25729E-05	0.26117E+02	-0.77343E+02	0.96306E+02	-0.46736E+02	-0.144E-02	0.659E-03	0.126E-03	0.242E-03
12	21	0.34299E-03	0.12052E-05	0.27481E+02	-0.81924E+02	0.10237E+03	-0.49498E+02	-0.697E-03	0.843E-03	-0.107E-04	0.929E-04
13	23	0.14400E-03	0.72443E-06	0.28685E+02	-0.85797E+02	0.10725E+03	-0.51594E+02	-0.224E-03	0.689E-03	-0.429E-03	-0.128E-03
14	25	0.14845E-03	0.71197E-06	0.28916E+02	-0.86679E+02	0.10855E+03	-0.52246E+02	-0.192E-03	0.685E-03	-0.432E-03	-0.138E-03
15	27	0.43064E-03	0.47545E-06	0.36093E+02	-0.11405E+03	0.14890E+03	-0.72381E+02	0.224E-03	0.253E-03	-0.331E-03	0.301E-03
16	29	0.34003E-03	0.40703E-06	0.34528E+02	-0.10818E+03	0.14037E+03	-0.68171E+02	0.206E-03	0.445E-03	-0.299E-03	0.278E-03
17	31	0.63374E-04	0.38902E-06	0.34313E+02	-0.10719E+03	0.13870E+03	-0.67247E+02	0.157E-03	0.305E-03	-0.471E-03	0.223E-03
18	32	0.14503E-03	0.30074E-06	0.35067E+02	-0.11034E+03	0.14369E+03	-0.69884E+02	0.854E-04	0.246E-03	-0.478E-03	0.673E-04
19	34	0.59585E-04	0.26184E-06	0.34631E+02	-0.10878E+03	0.14149E+03	-0.68817E+02	-0.670E-05	0.279E-03	-0.376E-03	0.206E-03
20	37	0.91125E-04	0.26054E-06	0.35094E+02	-0.11054E+03	0.14407E+03	-0.70110E+02	0.437E-04	0.275E-03	-0.367E-03	0.219E-03
21	39	0.21647E-03	0.20172E-06	0.35750E+02	-0.11341E+03	0.14878E+03	-0.72662E+02	0.733E-05	0.354E-03	-0.205E-03	0.186E-03
22	41	0.24074E-03	0.20096E-06	0.35869E+02	-0.11388E+03	0.14949E+03	-0.73022E+02	0.408E-05	0.347E-03	-0.194E-03	0.207E-03
23	42	0.18256E-03	0.18373E-06	0.36052E+02	-0.11456E+03	0.15048E+03	-0.73509E+02	-0.419E-05	0.303E-03	-0.227E-03	0.201E-03
24	43	0.15020E-03	0.17917E-06	0.36163E+02	-0.11496E+03	0.15103E+03	-0.73773E+02	-0.434E-05	0.275E-03	-0.251E-03	0.202E-03
25	44	0.19623E-03	0.13847E-06	0.37770E+02	-0.12126E+03	0.16057E+03	-0.78647E+02	0.375E-04	0.227E-03	-0.194E-03	0.220E-03

```
 26  45   0.20238E-03   0.13726E-06   0.37821E+02  -0.12146E+03   0.16088E+03   -0.78814E+02 0.352E-04 0.226E-03-0.186E-03 0.223E-03
 27  46   0.21285E-03   0.12440E-06   0.37860E+02  -0.12173E+03   0.16142E+03   -0.79141E+02-0.210E-04 0.221E-03-0.154E-03 0.227E-03
 28  48   0.20564E-03   0.92980E-07   0.38740E+02  -0.12541E+03   0.16727E+03   -0.82251E+02-0.957E-05 0.280E-03-0.582E-04 0.104E-03
 29  50   0.16682E-03   0.91745E-07   0.38489E+02  -0.12445E+03   0.16586E+03   -0.81542E+02-0.304E-04 0.282E-03-0.717E-04 0.801E-04
7
 30  51   0.10854E-03   0.62807E-07   0.39089E+02  -0.12686E+03   0.16955E+03   -0.83450E+02-0.940E-04 0.199E-03-0.775E-04 0.905E-04

 31  53   0.62574E-04   0.52257E-07   0.39594E+02  -0.12865E+03   0.17205E+03   -0.84642E+02-0.475E-04 0.132E-03-0.108E-03 0.116E-03
 32  55   0.51018E-04   0.49658E-07   0.39993E+02  -0.13023E+03   0.17444E+03   -0.85870E+02-0.724E-04 0.898E-04-0.106E-03 0.134E-03
 33  56   0.33198E-04   0.41276E-07   0.40410E+02  -0.13184E+03   0.17686E+03   -0.87101E+02-0.527E-04 0.761E-04-0.137E-03 0.118E-03
 34  58   0.10450E-03   0.35515E-07   0.40906E+02  -0.13380E+03   0.17986E+03   -0.88649E+02-0.255E-04 0.953E-04-0.981E-04 0.134E-03
 35  59   0.81217E-04   0.20629E-07   0.41496E+02  -0.13618E+03   0.18355E+03   -0.90584E+02-0.231E-04 0.960E-04-0.597E-04 0.855E-04
 36  61   0.98119E-04   0.20219E-07   0.41744E+02  -0.13713E+03   0.18499E+03   -0.91315E+02-0.140E-04 0.895E-04-0.508E-04 0.971E-04
7
 37  63   0.10435E-03   0.19285E-07   0.41758E+02  -0.13725E+03   0.18523E+03   -0.91468E+02-0.286E-04 0.101E-03-0.[illegible]E-04 0.860E-04
 38  65   0.10549E-03   0.18578E-07   0.41849E+02  -0.13759E+03   0.18574E+03   -0.91724E+02-0.754E-05 0.111E-03-0.[illegible]E-04 0.731E-04
 39  66   0.74156E-04   0.13812E-07   0.42038E+02  -0.13833E+03   0.18686E+03   -0.92298E+02-0.[illegible]E-05 0.954E-04-0.[illegible]E-04 0.563E-04
 40  69   0.74206E-04   0.97392E-08   0.44212E+02  -0.14683E+03   0.19974E+03   -0.98914E+02-0.253E-04-0.195E-04 0.321E-05 0.932E-04
 41  70   0.57316E-05   0.45290E-08   0.43496E+02  -0.14401E+03   0.19545E+03   -0.96701E+02-0.245E-04-0.566E-05-0.460E-04 0.422E-04
 42  72   0.22994E-04   0.43590E-08   0.43522E+02  -0.14412E+03   0.19563E+03   -0.96797E+02-0.287E-04-0.979E-05-0.500E-04 0.307E-04
7
 43  73   0.19892E-05   0.33545E-08   0.43444E+02  -0.14388E+03   0.19534E+03   -0.96682E+02-0.429E-04 0.956E-05-0.244E-04 0.287E-04
 44  74   0.30781E-04   0.13981E-08   0.43947E+02  -0.14579E+03   0.19816E+03   -0.98109E+02 0.153E-04 0.242E-04-0.991E-05 0.219E-04
 45  75   0.62087E-05   0.13173E-08   0.43767E+02  -0.14513E+03   0.19723E+03   -0.97653E+02 0.134E-05 0.343E-04-0.105E-04-0.524E-05
 46  76   0.12787E-04   0.58200E-09   0.43925E+02  -0.14574E+03   0.19814E+03   -0.98116E+02-0.333E-05 0.203E-04-0.830E-05 0.953E-05
 47  78   0.16871E-04   0.55845E-09   0.43980E+02  -0.14596E+03   0.19848E+03   -0.98291E+02-0.261E-05 0.210E-04-0.395E-05 0.965E-05
 48  79   0.13342E-04   0.34397E-09   0.44083E+02  -0.14637E+03   0.19910E+03   -0.98612E+02-0.650E-05 0.134E-04-0.285E-05 0.106E-04
7
 49  80   0.91758E-05   0.13009E-09   0.44281E+02  -0.14713E+03   0.20026E+03   -0.99203E+02-0.384E-05 0.434E-05-0.162E-05 0.969E-05
 50  81   0.73374E-05   0.85527E-10   0.44349E+02  -0.14739E+03   0.20063E+03   -0.99391E+02 0.315E-05 0.332E-05-0.311E-05 0.741E-05
 51  82   0.43006E-05   0.69728E-10   0.44312E+02  -0.14725E+03   0.20042E+03   -0.99284E+02 0.291E-05 0.571E-05-0.398E-05 0.358E-05
 52  83   0.16525E-05   0.75653E-11   0.44401E+02  -0.14760E+03   0.20096E+03   -0.99568E+02 0.292E-06 0.203E-05-0.116E-05 0.143E-05
 53  85   0.54000E-06   0.67224E-11   0.44401E+02  -0.14760E+03   0.20096E+03   -0.99567E+02 0.107E-06 0.146E-05-0.180E-05 0.115E-05
 54  86   0.38198E-06   0.87947E-12   0.44473E+02  -0.14789E+03   0.20140E+03   -0.99795E+02 0.171E-06-0.358E-06 0.521E-06-0.671E-06
SOLUTION FOUND
 54  86   0.38198E-06   0.87947E-12   0.44473E+02  -0.14789E+03   0.20140E+03   -0.99795E+02 0.171E-06-0.358E-06 0.521E-06-0.671E-06
SUM OF RESIDUALS SQUARED= 0.87947E-12
THE MODIFIED LAGRANGIAN MULTIPLIERS ARE          -0.390440403E+01   0.444731142E+02   -0.147886173E+03   0.201402717E+03
                                                 -0.997951083E+02
LAGRANGIAN MULTIPLIERS
-0.1472E+04  0.7707E+03 -0.1509E+03  0.1309E+02 -0.4251E+00
LOG OF CURRENT CYCLES, LOG XNM
  0.5723E+01  0.9638E+01  0.5919E+01  0.6115E+01  0.6311E+01
  0.6506E+01  0.6702E+01  0.6898E+01  0.7094E+01  0.7289E+01
  0.7485E+01  0.7681E+01  0.7876E+01  0.8072E+01  0.8268E+01
  0.8464E+01  0.8659E+01  0.8855E+01  0.9051E+01  0.9246E+01
  0.9442E+01
ORDERED LOG OF CURRENT CYCLES, LOG,XNM,                 X AXIS PDF, CDF PLOT
  0.5723E+01  0.5919E+01  0.6115E+01  0.6311E+01  0.6506E+01
  0.6702E+01  0.6898E+01  0.7094E+01  0.7289E+01  0.7485E+01
  0.7681E+01  0.7876E+01  0.8072E+01  0.8268E+01  0.8464E+01
  0.8659E+01  0.8855E+01  0.9051E+01  0.9246E+01  0.9442E+01
  0.9638E+01
GCDF PARAMETERS
  2
CDF OF LOG OF CURRENT CYCLES, LOG XNM,                  Y AXIS OF PDF, CDF PLOT
  0.0000E+00  0.3801E-02  0.1898E-01  0.5763E-01  0.1267E+00
  0.2208E+00  0.3259E+00  0.4288E+00  0.5219E+00  0.6038E+00
  0.6756E+00  0.7398E+00  0.7981E+00  0.8515E+00  0.8992E+00
  0.9391E+00  0.9689E+00  0.9874E+00  0.9963E+00  0.9993E+00
  0.1000E+01
```

ORIGINAL PAGE IS OF POOR QUALITY

```
                                                              X AXIS OF PDF, CDF PLOT
ORDERED LOG OF CURRENT CYCLES, LOG XNM.
 0.5723E+01   0.5919E+01   0.6115E+01   0.6311E+01   0.6506E+01
 0.6702E+01   0.6898E+01   0.7094E+01   0.7289E+01   0.7485E+01
 0.7681E+01   0.7876E+01   0.8072E+01   0.8268E+01   0.8464E+01
 0.8659E+01   0.8855E+01   0.9051E+01   0.9246E+01   0.9442E+01
 0.9638E+01
 0.5723E+01   0.5919E+01   0.6115E+01   0.6311E+01   0.6506E+01
 0.6702E+01   0.6898E+01   0.7094E+01   0.7289E+01   0.7485E+01
 0.7681E+01   0.7876E+01   0.8072E+01   0.8268E+01   0.8464E+01
 0.8659E+01   0.8855E+01   0.9051E+01   0.9246E+01   0.9442E+01
 0.9638E+01
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
YYYYYYYYYY   3333333333333333333333333333333333333333333333333333333333333333333333333333333333333333   YYYYYYYYYY
YYYYYYYYYY   3333333333333333333333333  Digital Equipment Corporation - VAX/VMS Version V4.7  333333333333333333333333   YYYYYYYYYY
YYYYYYYYYY   3333333333333333333333333333333333333333333333333333333333333333333333333333333333333333   YYYYYYYYYY
```

NETNONPRIV

PLOT1

CPR;1

File _DBA0:[]PLOT1.CPR;1 (359,204,0), last revised on 23-NOV-1988 11:20, is a 2 block sequential file owned by UIC [11,11]. The records are variable length with FORTRAN (FTN) carriage control. The longest record is 25 bytes.

Job PLOT1 (683) queued to SYS$SBPRT on 23-NOV-1988 11:20 by user NETNONPRIV, UIC [11,11], under account 20100ADD at priority 100, started on printer _TTF7: on 23-NOV-1988 11:20 from queue TTF7.

```
YYYYYYYYYY   3333333333333333333333333333333333333333333333333333333333333333333333333333333333333333   YYYYYYYYYY
YYYYYYYYYY   3333333333333333333333333  Digital Equipment Corporation - VAX/VMS Version V4.7  333333333333333333333333   YYYYYYYYYY
YYYYYYYYYY   3333333333333333333333333333333333333333333333333333333333333333333333333333333333333333   YYYYYYYYYY
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
(E12.4,1X,E12.4)
 0.5723E+01   0.5149E-02
 0.5919E+01   0.3369E-01
 0.6115E+01   0.1213E+00
 0.6311E+01   0.2735E+00
 0.6506E+01   0.4324E+00
 0.6702E+01   0.5291E+00
 0.6898E+01   0.5448E+00
 0.7094E+01   0.5056E+00
 0.7289E+01   0.4464E+00
 0.7485E+01   0.3897E+00
 0.7681E+01   0.3444E+00
 0.7876E+01   0.3110E+00
 0.8072E+01   0.2852E+00
 0.8268E+01   0.2600E+00
 0.8464E+01   0.2272E+00
 0.8659E+01   0.1810E+00
 0.8855E+01   0.1230E+00
 0.9051E+01   0.6575E-01
 0.9246E+01   0.2512E-01
 0.9442E+01   0.6140E-02
 0.9638E+01   0.8464E-03
```

```
KKKKKKKKKK    2222222222222222222222222222222222222222222222222222222222222222222222222222222222222222222222    KKKKKKKKKK
KKKKKKKKKK    222222222222222222222222  Digital Equipment Corporation - VAX/VMS Version V4.7  2222222222222222222222222    KKKKKKKKKK
KKKKKKKKKK    2222222222222222222222222222222222222222222222222222222222222222222222222222222222222222222222    KKKKKKKKKK
```

NETNONPRIV

PLOT2

.CPR;1

ORIGINAL PAGE IS
OF POOR QUALITY

```
File _DBA0:[]PLOT2.CPR;1 (363,197,0), last revised on 23-NOV-1988 11:21, is a 2 block sequential file owned by UIC [11,11].  The
records are variable length with FORTRAN (FTN) carriage control.  The longest record is 25 bytes.

Job PLOT2 (684) queued to SYS$$PRT on 23-NOV-1988 11:21 by user NETNONPRIV, UIC [11,11], under account 20100ADD at priority 100,
started on printer _TTF6: on 23-NOV-1988 11:21 from queue TTF6.
```

```
KKKKKKKKKK    2222222222222222222222222222222222222222222222222222222222222222222222222222222222222222222222    KKKKKKKKKK
KKKKKKKKKK    222222222222222222222222  Digital Equipment Corporation - VAX/VMS Version V4.7  2222222222222222222222222    KKKKKKKKKK
KKKKKKKKKK    2222222222222222222222222222222222222222222222222222222222222222222222222222222222222222222222    KKKKKKKKKK
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
(E12.4,1X,E12.4)
 0.5723E+01   0.0000E+00
 0.5919E+01   0.3801E-02
 0.6115E+01   0.1898E-01
 0.6311E+01   0.5763E-01
 0.6506E+01   0.1267E+00
 0.6702E+01   0.2208E+00
 0.6898E+01   0.3259E+00
 0.7094E+01   0.4288E+00
 0.7289E+01   0.5217E+00
 0.7485E+01   0.6038E+00
 0.7681E+01   0.6756E+00
 0.7876E+01   0.7398E+00
 0.8072E+01   0.7981E+00
 0.8268E+01   0.8515E+00
 0.8464E+01   0.8992E+00
 0.8659E+01   0.9391E+00
 0.8855E+01   0.9689E+00
 0.9051E+01   0.9874E+00
 0.9246E+01   0.9963E+00
 0.9442E+01   0.9993E+00
 0.9638E+01   0.1000E+01
```

9.0 APPENDIX D

IMSL SUBROUTINE CALLS FROM RANDOM3 AND RANDOM4

RANDOM3

1. RNSET - Initializes a random seed for use in the IMSL random number generators.
2. RNNOR - Generates pseudorandom numbers from a standard normal distribution using an inverse CDF method.
3. RNLNL - Generates pseudorandom numbers from a lognormal distribution.
4. DESPL - Performs nonparametric probability density function estimation by the penalized likelihood method.
5. GCDF - Evaluates a general continuous cumulative distribution function given the ordinates of the density.

RANDOM4

1. RNSET - Initializes a random seed for use in the IMSL random number generators.
2. RNNOR - Generates pseudorandom numbers from a standard normal distribution using an inverse CDF method.
3. RNLNL - Generates pseudorandom numbers from a lognormal distribution.

10.0 APPENDIX E

SAMPLE SAS/GRAPH PROGRAM FOR RANDOM3 AND RANDOM4

```
data a;
INFILE 'PLOT1.CPR' FIRSTOBS=2;input x y;
GOPTIONS DEVICE=HP7470;
proc gplot;
   axis1 label=(h=1 f=simplex 'LOG OF CYCLES')
         value=(h=1 f=simplex);
   axis2 value=(h=1 f=simplex) label=none;
   plot y*x / haxis=axis1 vaxis=axis2;
   TITLE H=1 A=90 F=SIMPLEX 'PROBABILITY DENSITY FUNCTION';
   symbol i=spline v=square;
data B;
INFILE 'PLOT2.CPR' FIRSTOBS=2;input x y;
proc gplot;
   axis1 label=(h=1 f=simplex 'LOG OF CYCLES')
         value=(h=1 f=simplex);
   axis2 value=(h=1 f=simplex) label=none;
   plot y*x / haxis=axis1 vaxis=axis2;
   TITLE H=1 A=90 F=SIMPLEX 'CUMULATIVE DISTRIBUTION FUNCTION';
   symbol i=spline v=square;
```